e2v

ELIIXA+ 16k

Cmos Multi-Line Monochrome Camera

User Manual



Summary

1 CAMERA OVERVIEW

1.1 Features

- Cmos Sensor 4x 16384 Pixels, 5 x 5µm
- Multi-Line structure (1, 2 or 4 lines to adapt the sensitivity)
- Interface :
  - Medium/Full CameraLink®, 85MHz per Channel
  - Full+ (10 Taps) CameraLink®, 85MHz per Channel, for "**BA1**" versions only
- Line Rate :
  - Up to 50000 l/s In CameraLink® for "**BA1**" versions
  - Up to 40000 l/s In CameraLink® for "**BA0**" versions
- Data Rate :
  - Up to 850 MB/s In CameraLink®
- Bit Depth : 8 or 12bits
- Flat Field Correction
- Look Up Table
- Low Power Consumption : <13W
- Compliant with Standard Lenses of the Market
- Full Exposure Control, even in 4S "TDE" mode

1.2 Key Specifications

Note : All values in LSB is given in 12 bits format

Characteristics	**Typical Value**	**Unit**
Sensor Characteristics at Maximum Pixel Rate		
Resolution	4 x 16384	Pixels
pixel size (square)	5 x 5	µm
Max line rate – CameraLink Full+ 10 x 85MHz	50 **(*)**	kHz
Max line rate – CameraLink Full 8 x 85MHz	40	kHz
Max line rate – CameraLink Medium 4 x 85MHz	20	kHz
Radiometric Performance at Maximum Pixel Rate and minimum camera gain		
Bit depth	8, 12	Bits
Response (broadband)	450	LSB/(nJ/cm²)
Full Well Capacity	27300 (in 2S or 4S mode and MultiGain at 1/2)	electrons
Response non linearity	0,3	%
PRNU HF Max	3	%
Dynamic range (1S / 2S / 4S mode)	67,6 / 70,7 / 68,7	dB

(*) Not available for EV71YC4MCL1605-**BA0** versions

Functionality (Programmable via Control Interface)		
Analog Gain	Up to 12 (x4)	dB
Offset	-4096 to +4096	LSB
Trigger Mode	Timed (Free run) and triggered (Ext Trig, Ext ITC) modes	
Mechanical and Electrical Interface		
Size (w x h x l)	100 x 156 x 36	mm
Weight	700	g
Lens Mount	M95 x 1	-
Sensor alignment (see chapter 4)	±100	µm
Sensor flatness	±35	µm
Power supply	Single 12 DC to 24 DC	V
Power dissipation - CameraLink	< 13	W
General Features		
Operating temperature	0 to 55 (front face) or 70 (Internal)	°C
Storage temperature	-40 to 70	°C
Regulatory	CE, FCC and RoHS compliant	

1.3 Description

e2v's next generation of line scan cameras are setting new, high standards for line rate and image quality. Thanks to e2v's recently developed multi line CMOS technology, the camera provides an unmatched 100 000 lines/s in a 16k pixel format and combines high response with an extremely low noise level; this delivers high signal to noise ratio even when short integration times are required or when illumination is limited. The 5µm pixel size is arranged in four active lines, ensuring optimal spatial resolution in both scanning and sensor directions with off-the-shelf lenses. This Camera opens up an array of new possibilities for the next generation of inspection systems for demanding applications such as flat panel display, PCB and solar cell inspection.

1.4 Typical Applications

- Flat Panel Display Inspection
- PCB Inspection
- Solar Cell Inspection
- Glass Inspection
- Print Inspection

2 CAMERA PERFORMANCES

2.1 Camera Characterization

	Unit	Mode 1S (0dB)			Mode 2S (0dB)			Mode 4S (0dB)		
		Min	Typ.	Max	Min	Typ.	Max	Min	Typ.	Max
Dark Noise RMS	LSB	-	1,7	2,2		2,4	3,1		3	4
Dynamic Range	-	-	2394:1	-	-	3412:1(*)	-	-	2730:1(*)	-
Readout Noise	e-	-	5,7	-	-	8	-	-	10	-
Full Well Capacity	e-	-	13650	-	-	27300	-	-	27300	-
SNR	dB	-	40	-	-	43(*)	-	-	43(*)	-
Peak Response (660nm)	LSB/ (nJ/cm2)	-	137	-	-	274	-	-	547	-
Non Linearity	%	-	0,3	-	-	0,3	-	-	0,3	-
Without Flat Field Correction :										
FPN rms	LSB	-	0,4	1,5	-	0,7	1,5	-	0,8	1,5
FPN pk-pk	LSB	-	3,2	15	-	5	15	-	5,6	15
PRNU hf (3/4 Sat)	%	-	0,13	0,25	-	0,1	0,25	-	0,1	0,25
PRNU pk-pk (3/4 Sat)	%	-	1	3	-	0,8	3	-	0,8	3

Test conditions :

- Figures in LSB are for a 12bits format.
- Measured at exposure time = 50µs and line period = 50µs in Ext Trig Mode (Max Exposure Time)
- Maximum data rate
- Stabilized temperature 30/40/55 °C (Room/Front Face/Internal)
- SNR Calculated at 75% Vsat with minimum Gain.

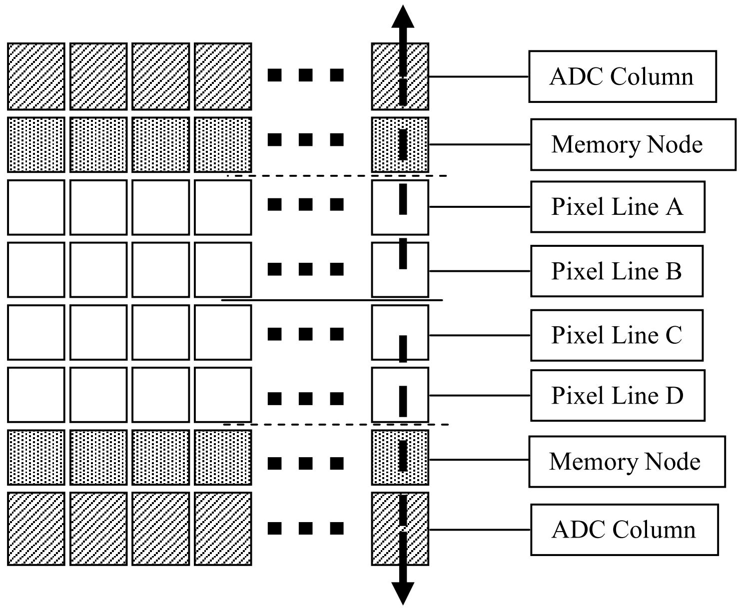
(*) In mode 2S/4S, only with the use of the Multi-Line Gain

2.2 Image Sensor

The Eliixa+ 16k sensor is composed of two pairs of sensitive lines. Each pair of lines use the same Analog to Digital Column converter (ADC Column). An appropriate (embedded) Time delay in the exposure between each line this allows to combine two successive exposures in order to double the sensitivity of a single line.

This Time Delay Exposure is used only in the 4S multi-line modes (4 Lines) as described below.

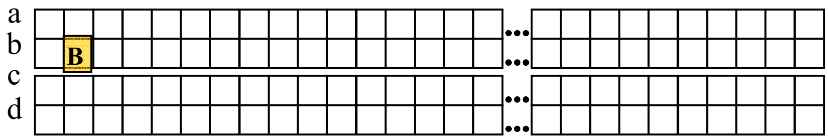
The 16384 Pixels of the whole sensor are divided in 4 blocks of 4096 pixels.



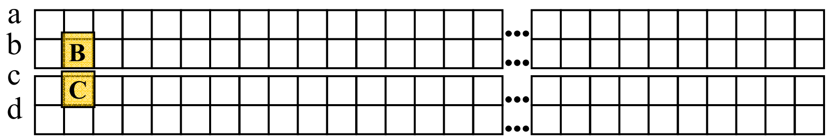
2.3 Multi-Lines modes

Multi-Lines Modes ***(16k Pixels Output)***

Mode 1S = B

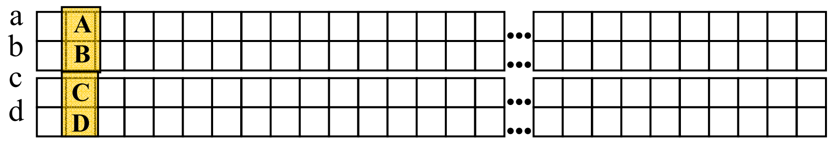


Mode 2S = B+C (FPGA)



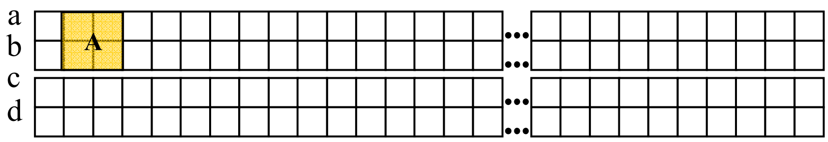
Mode 4S = (A.B)+(C.D)

Note : (A.B) = summation in the sensor

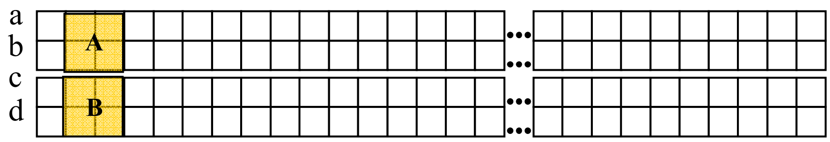


Binning Modes ***(8k Pixels Output) :*** *Not available on EV71YC4MCL1606-**BA0** versions*

Mode 1SB = A

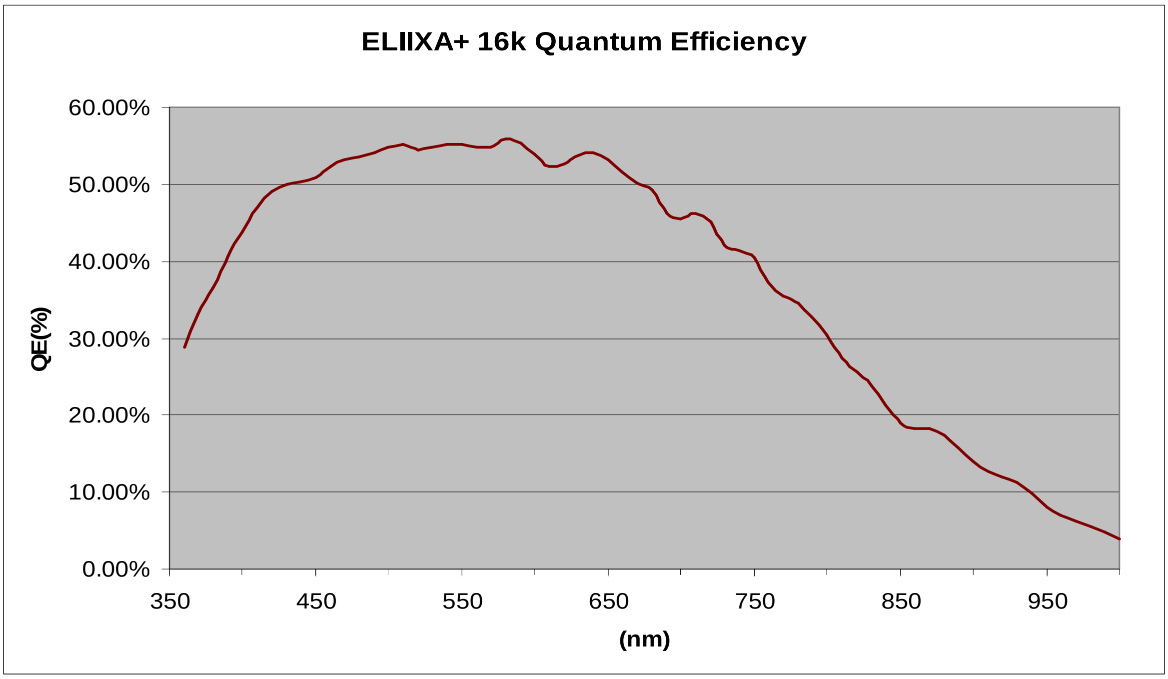


Mode 2SB = (A+B)

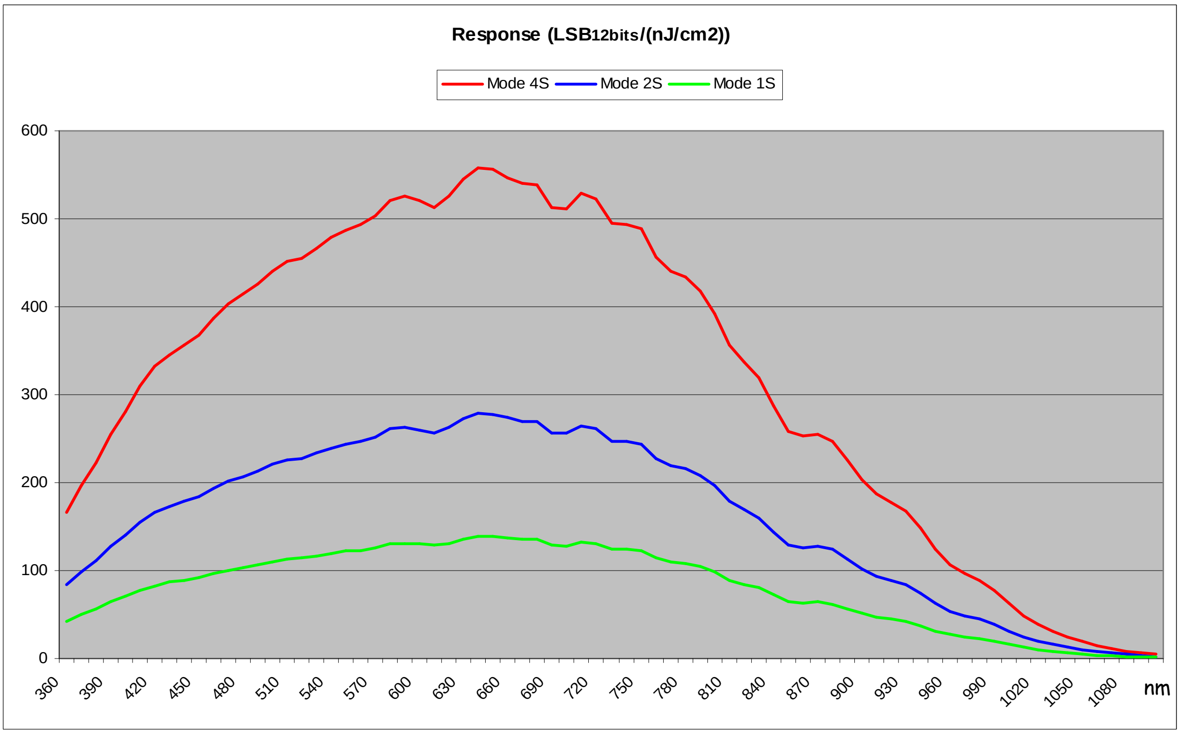


2.4 Response & QE curves

2.4.1 Quantum Efficiency

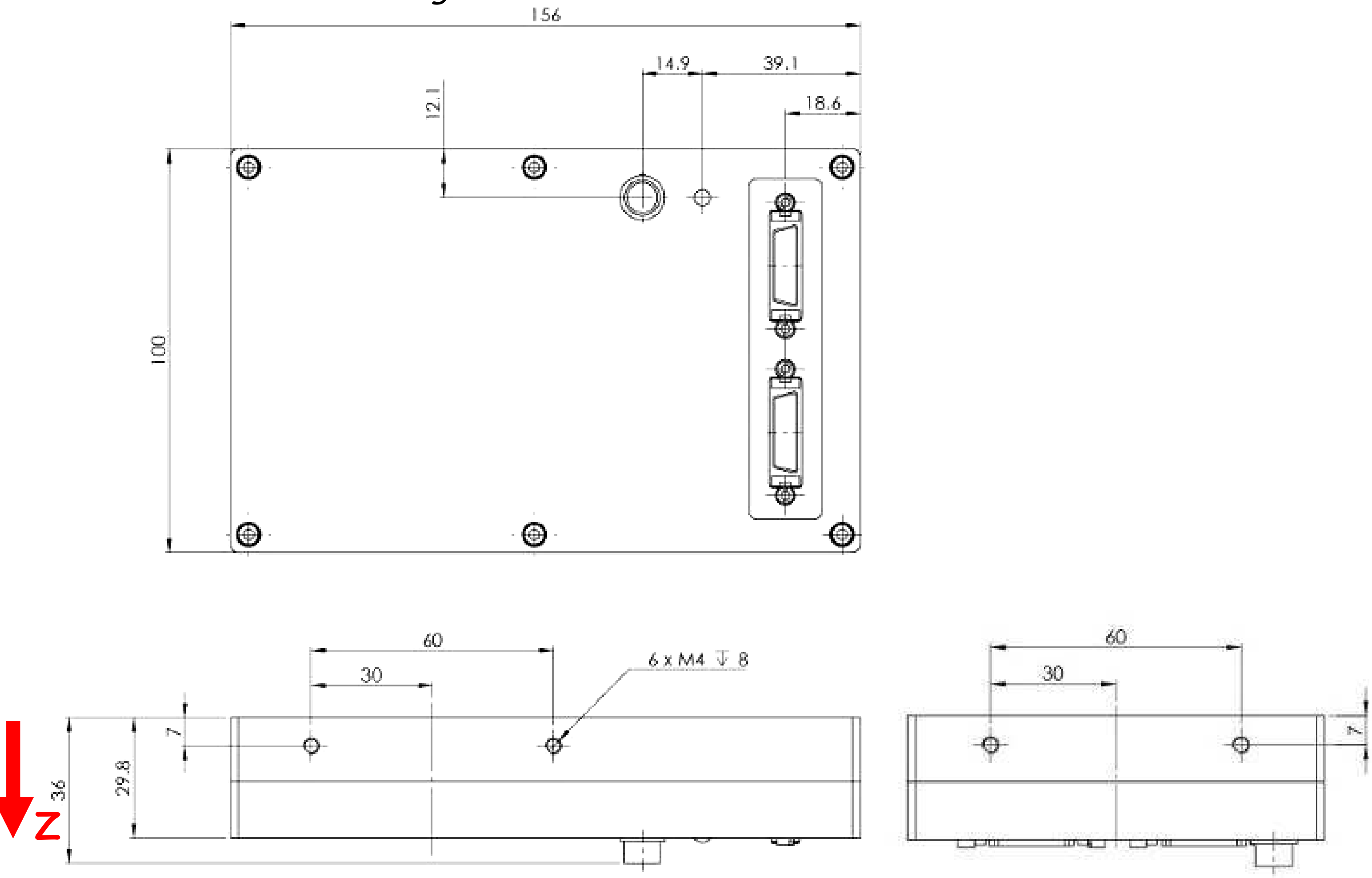


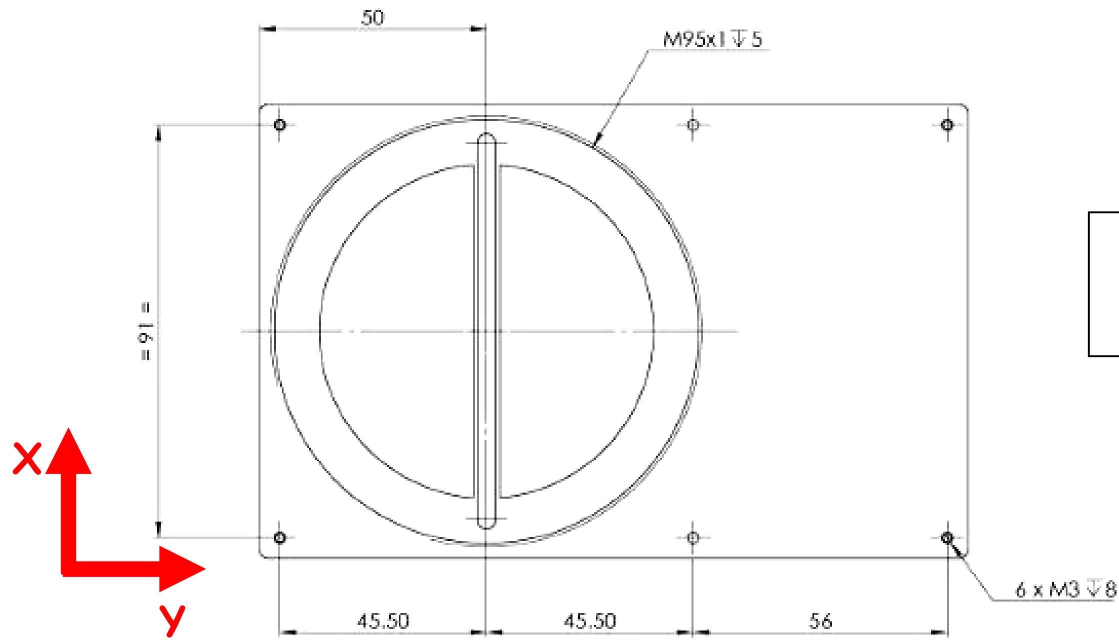
2.4.2 Spectral Response



3 CAMERA HARDWARE INTERFACE

3.1 *Mechanical Drawings*

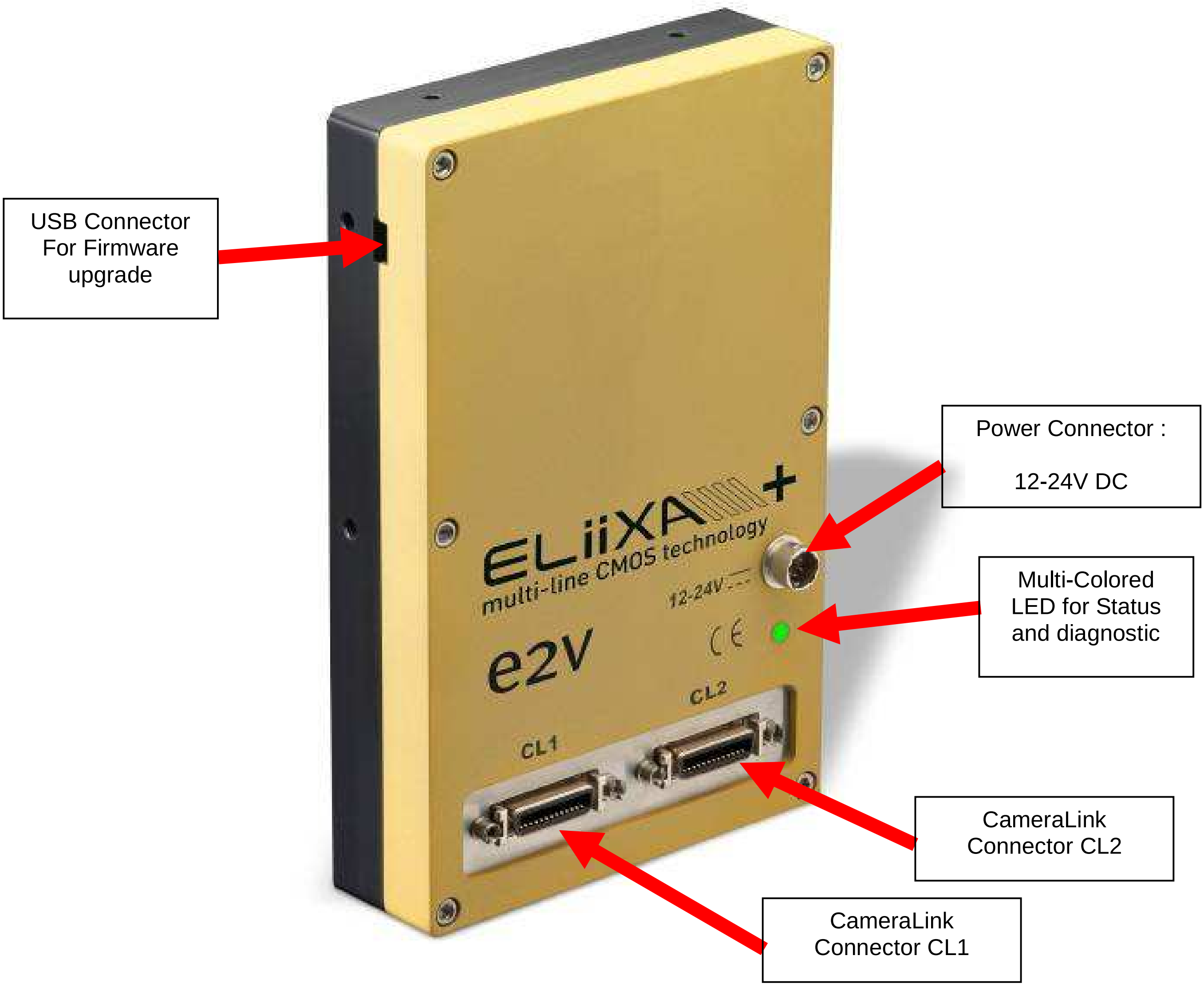




The Step file is available on the web : www.e2v.com/cameras

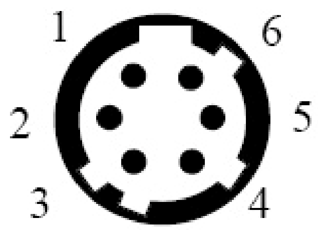
Sensor alignment	
Z = -9.4 mm	±100µm
X = 9 mm	±100 µm
Y = 50mm	±100 µm
Flatness	±25 µm
Rotation (X,Y plan)	±0,1°
Tilt (versus lens mounting plane)	50µm

3.2 Input/output Connectors and LED



3.2.1 Power Connector

Camera connector type: Hirose HR10A-7R-6PB (male)
Cable connector type: Hirose HR10A-7P-6S (female)



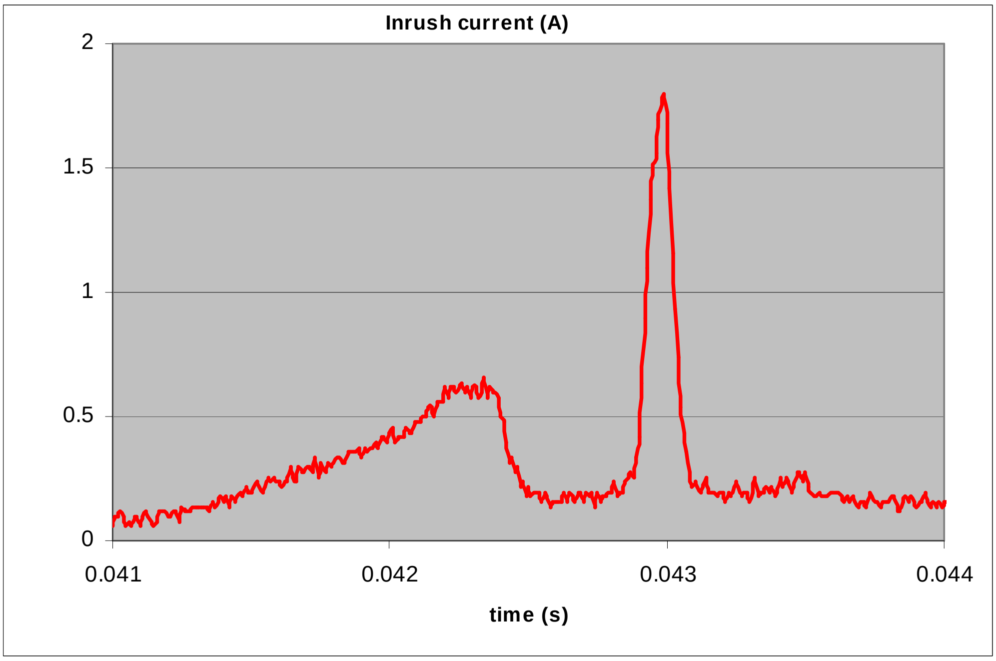
Camera side description

Signal	Pin	Signal	Pin
PWR	1	GND	4
PWR	2	GND	5
PWR	3	GND	6

Power supply from 12 to 24v
Power 13W max with an typical inrush current peak of **1,8A** during power up

Typical values	Current consumption	
	12V	**24V**
ELIIXA+ CL (normal)	1,06A	0,54A
ELIIXA+ CL (Standby)	0,47A	0,25A

Power up Time : Around 43s (Green Light)



3.2.2 Status LED Behaviour

After less than 2 seconds of power establishment, the LED first lights up in ORANGE. Then after a Maximum of 30 seconds, the LED must turn in a following colour :

Colour and state	Meaning
Green and continuous	OK
Green and blinking slowly	Waiting for Ext Trig (Trig1 and/or Trig2)
Red and continuous	Camera out of order : Internal firmware error

3.2.3 CameraLink Output Configuration

	Connector CL1 + CL2	Pixels per Channel
Medium CameraLink Mode		
4 Channels 8bits	4 x 85MHz	4 x 4096
4 Channels 12bits	4 x 85MHz	4 x 4096
Full CameraLink Mode		
8 Channels 8bits	8 x 85MHz	8 x 2048
Full + CameraLink Mode (not available for EV71YC4MCL1605-**BA0** versions)		
10 Channels 8bits	10 x 85MHz	10 x 1638

4 STANDARD CONFORMITY

The ELIIXA+ cameras have been tested using the following equipment:

- A shielded power supply cable
- A Camera Link data transfer cable ref. 14B26-SZLB-500-OLC(3M)

e2v recommends using the same configuration to ensure the compliance with the following standards.

4.1 CE Conformity

The ELIIXA+ cameras comply with the requirements of the EMC (European) directive 2004/108/CE (EN50081-2, EN 61000-6-2).

4.2 FCC Conformity

The ELIIXA+ cameras further comply with Part 15 of the FCC rules, which states that: Operation is subject to the following two conditions:

- This device may not cause harmful interference, and
- This device must accept any interference received, including interference that may cause undesired operation

This equipment has been tested and found to comply with the limits for Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

Warning: Changes or modifications to this unit not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment.

4.3 RoHs Conformity

ELIIXA+ cameras comply with the requirements of the RoHS directive 2002/95/EC.

EU Declaration of Conformity

Declaration Number: NE31S208701

We,

e2v semiconductors
rue de Rochepleine
38120 Saint-Egrève
France

declare the product(s)

Product Family: ***EliiXA+ 16k Cameras***

Model Identification: **EV71YC4MCL1605-Bxx / EV71YC4MCP1605-Bxx**
x = 0-9-A-Z

---

in conformance with the requirements of the following standards:

EN55022 : ed. 2006, A class

EN61000-6-2 : ed. 2005

IEC 61000-4-2	: ed.2009
IEC 61000-4-3	: ed.2006 + A1/2008 +A2/2011
IEC 61000-4-4	: ed.2004
IEC 61000-4-5	: ed.2006
IEC 61000-4-6	: ed.2009
IEC 61000-4-11	: ed.2004

when used in conformity with the recommended set-up (as per the Product Specification or Data Sheet).

---

applicable to: **Information Technology Equipments (I.T.E.)**

This (These) product(s) complies(y) with the requirements of the:

- Electromagnetic Compatibility Directive 2004/108/EC,
- CE Marking European Directive 93/68/EEC

and carry the CE marking accordingly.

Saint-Egrève, France, on August 23rd, 2012



Martine WOOLF,
Quality Manager

Printed form [illegible]

e2v semiconductors • Avenue de Rochepleine • BP 123 • 38521 Saint Egreve Cedex • France

5 GETTING STARTED

5.1 Out of the box

The contains of the Camera box is the following :

- One Camera ELIIXA+
- Power connector (Hirose HR10A-7P-6S -female)

There is no CDROM delivered with the Camera : Both User Manual (this document) and CommCam control software have to be downloaded from the web site : This ensure you to have an up-to-date version.

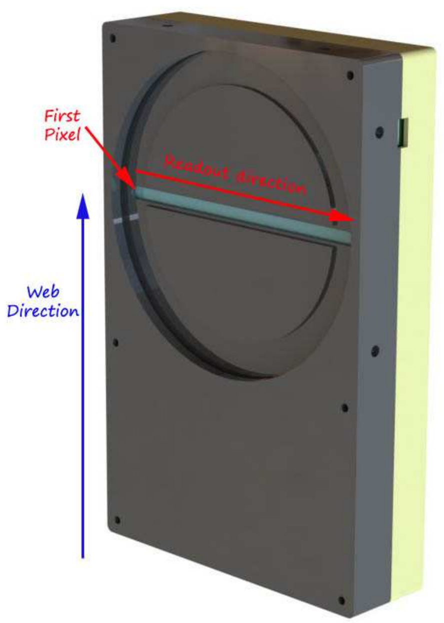
Main Camera page : ***www.e2v.com/cameras***

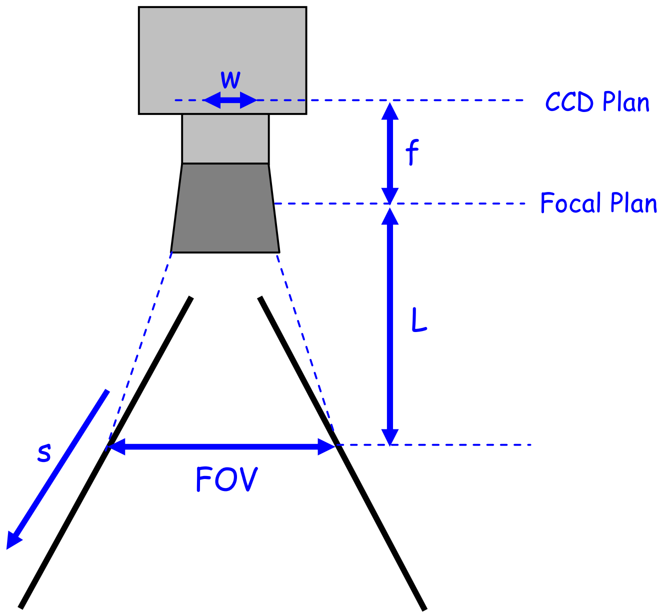
On the appropriate Camera Page (ELIIXA+) you'll find a download link first version of CommCam compliant is indicated in the last Chapter

CommCam download requires a login/password :

- *Login :* ***commcam***
- *Password :* ***chartreuse***

5.2 Setting up in the system





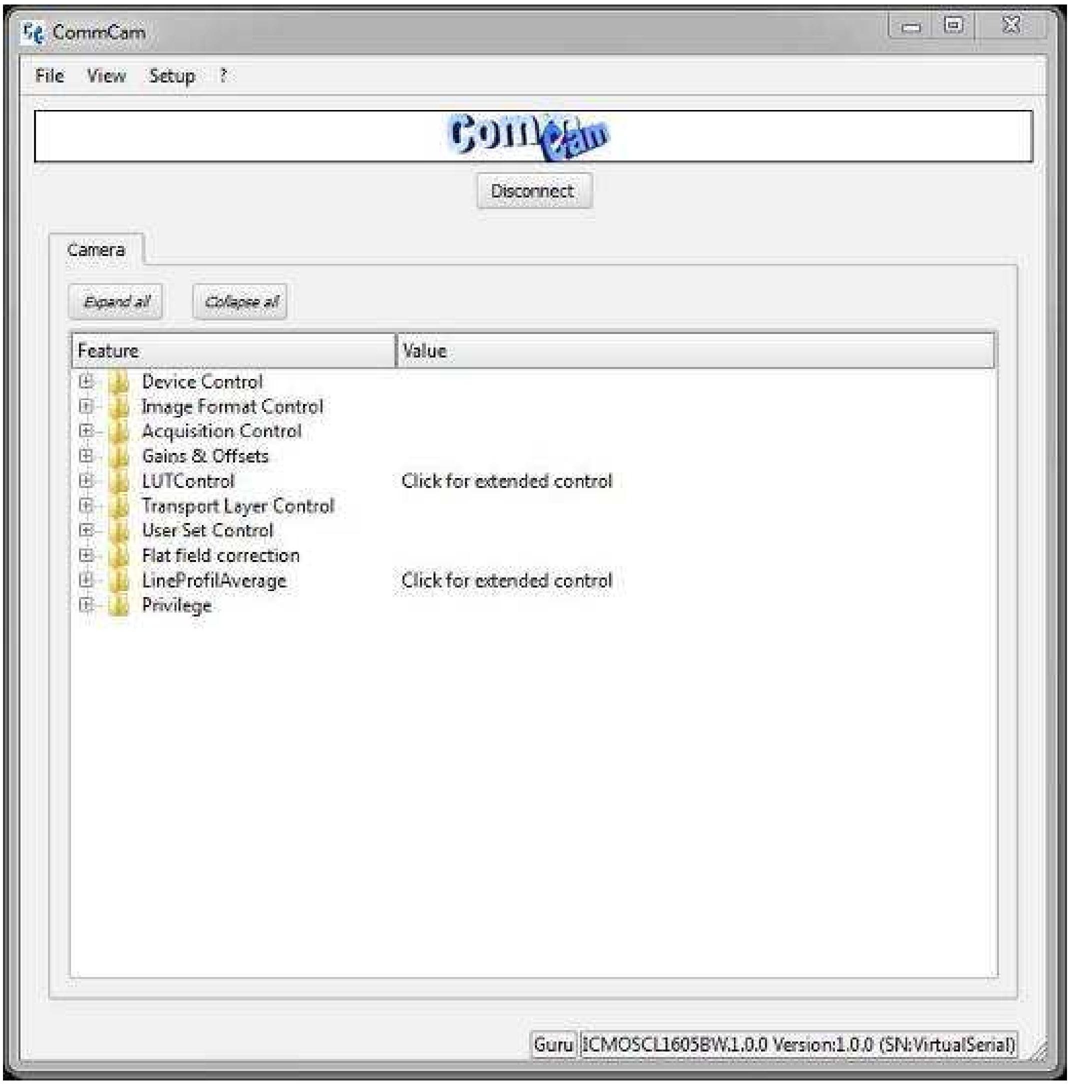
$$\frac{w}{FOV} = \frac{f}{L}$$

The Compliant Lenses and their accessories are detailed in Appendix C

6 CAMERA SOFTWARE INTERFACE

6.1 Control and Interface

As all the e2v Cameras, the ELIIXA+ CL is delivered with the friendly interface control software COMMCAM.UCL (as "Ultimate Camera Link") which is based on the GenICam standard
COMMCAM recognizes and detects automatically all the UCL Cameras connected on any transport layers (Camera Link or COM ports) of your system.
Once connected to the Camera you have an easy access to all its features. The visibility of these features can be associated to three types of users: Beginner, Expert or Guru. Then you can make life easy for simple users.
Minimum version of CommCam is 2.1.0 in order to recognize the ELIIXA+ Camera ("BA0" version)



6.2 Serial Protocol and Command Format

The Camera Link interface provides two LVDS signal pairs for communication between the camera and the frame grabber. This is an asynchronous serial communication based on RS-232 protocol.
The serial line configuration is:

- Full duplex/without handshaking
- 9600 bauds (default), 8-bit data, no parity bit, 1 stop bit. The baud rate can be set up to 115200

6.2.1 Syntax

Internal camera configurations are activated by write or readout commands.
The command syntax for write operation is:

w <command_name> <command_parameters><CR>

The command syntax for readout operation is:

r <command_name><CR>

6.2.2 Command Processing

Each command received by the camera is processed:

- The setting is implemented (if valid)
- The camera returns ">"<return code><CR>

The camera return code has to be received before sending a new command.

> The camera return code has to be received before sending a new command. Some commands are longer than the others : Waiting for the return code ensure a good treatment of all the commands Without saturating the buffer of the camera

Table 5-1. Camera Returned Code

Returned code	meaning
>0	(or ">OK") : All right, the command will be implemented
>3	Error Bad CRC (for write command only)
>16	Invalid Command ID (Command not recognize or doesn't exist)
>33	Invalid Access (the receipt of the last command has failed).
>34	Parameter out of range (the parameter of the last command send is out of range).
>35	Access Failure (bad communication between two internal devices).

6.2.3 GenICam ready

The CameraLink Standard is not yet compliant with GenICam Standard, but as much as possible, each command of the ELIIXA+ will have its correspondence with the Standard Feature Naming Convention of the GenIcam Standard.
This correspondence is given in parenthesis for each feature/command as the following example :

- **Vendor name** (*DeviceVendorName*) : **"e2v"**

6.3 Camera Commands

6.3.1 Information

These values allow to indentify the Camera. They can be accessed in CommCam software in the "Info" section All these values are fixed in factory and can't be changed (shaded) except the Camera User ID which can be fixed by the Customer :

- **Vendor name** (*DeviceVendorName*) : "e2v"
  - ⇨ Read function : "**r vdnm**";
    Returned by the camera : "e2v", string of 32 bytes (including "/0")
  - ⇨ Can not be written
- **Model Name** (*DeviceModelName*) : Internal name for GenICam :
  - ⇨ Read function : "**r mdnm**";
    Returned by the camera : String of 32 bytes (including "/0") :
  - ⇨ Can not be written
- **Device Manufacturer Info** (*DeviceManufacturerInfo*) : Get Camera ID
  - ⇨ Read function : "**r idnb**";
    Returned by the camera : String of 128 bytes (including "/0")
  - ⇨ Can not be written
- **Device Version** (*DeviceVersion*) : Get Camera Hardware version
  - ⇨ Read function : "**r dhwv**";
    Returned by the camera : String of 32 bytes (including "/0")
  - ⇨ Can not be written
- **Device Firmware Version** (*DeviceFirmwareVersion*): Get camera synthetic firmware
  - ⇨ Read function : "**r dfwv**";
    Returned by the camera : String of 16 bytes (including "/0")
  - ⇨ Can not be written
- **Device SFNC Version : 1.5.0**
  These Parameters (Major, Minor, Sub Minor) are only virtual ones in order to give the SFNC compliance of the Camera.
- **Device ID** (*DeviceID*) : Camera Factory identifier ID
  - ⇨ Read function : "**r cust**";
    Returned by the camera : String of 128 bytes (including "/0")
  - ⇨ Write function : "**w cust <idstr>**"
- **Device User ID** (*DeviceUserID*) : Camera user identifier ID
  - ⇨ Read function : "**r cust**";
    Returned by the camera : String of 128 bytes (including "/0")
  - ⇨ Write function : "**w cust <idstr>**"

- **Electronic board ID** (*ElectronicBoardID*) : Get PcB Board ID
  - ⇨ Read function : "**r boid**";
    Returned by the camera : String of 32 bytes (including "/0")
  - ⇨ Can not be written

- **Device Temperature Selector** (*DeviceTemperatureSelector*) : **MainBoard**
  - ⇨ Can not be written

- **Device Temperature** (*DeviceTemperature*) : Get Main Board Temperature
  - ⇨ Read function : "**r temp**";
    Return by the camera : Temperature in Q10.2 format (8 bits signed + 2 bits below comma). Value is between -512 to 511 in °C.

- **Device Serial Port Selection** : Indicates the Serial Port on which the Camera is connected.

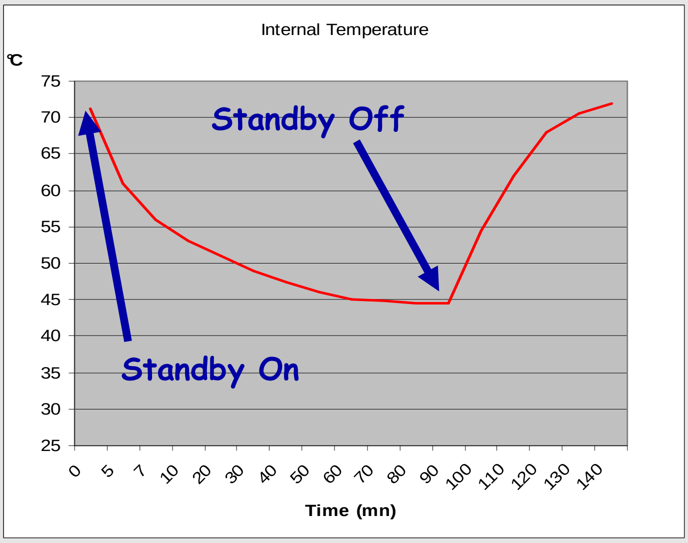
- **Device Serial Port Baud Rate** (*ComBaudRate*): Set the Camera BaudRate
  - ⇨ Read function : "**r baud**";
    Returned by the camera : Value of the Baud Rate
  - ⇨ Write function : "**w baud**" <index> with the index as follows :
    - 1 : 9600 Bauds (default value at power up)
    - 2 : 19200Bauds
    - 6 : 57600Bauds
    - 12 : 115200Bauds

- **Standby Mode** (*Standby*) : Activation of the Standby mode of the Camera
  - ⇨ Read function : "**r stby**";
    Returned by the camera : Boolean.
    - 0 : Disable Standby mode (False)
    - 1 : Enable stanby mode (True)
  - ⇨ Write function : "**w stby <val>**"; <val> is 0 or 1.

A standby mode, what for ?

The Standby mode stops all activity on the sensor level. The power dissipation drops down to about **6W**. During the standby mode, **the grab is stopped**

Once the Standby mode turned off, the Camera recovers in less than **1ms** to send images again from the sensor.



- **Camera status** : Get the Camera status register (32bits Integer)
  - ⇨ Read function : "**r stat**";
    Returned by the camera : 32bits integer :
    - **Bit 0** : (*StatusWaitForTrigger*) : True if no trig received from more than 1sec
    - **Bit 1** : (*StatusTriggerTooFast*) : Missing triggers. Trig signal too fast
    - **Bit 2** : (*StatusSensorConnection*) : True is the Sensor pattern is checked as failed.
    - Bit 3, 4, 5, 6, 7 : Reserved
    - **Bit 8** : (*StatusWarningOverflow*) : True is an overflow occurs during FFC or Tap balance processing.
    - **Bit 9** : (*StatusWarningUnderflow*) : True is an underflow occurs during FFC or Tap balance processing
    - Bits 10 : Reserved
    - **Bits 11** : Scrolling Direction : 0 = Forward, 1 = Reverse. Updated only by external CC3 (CameraLink)
    - Bits, 12, 13, 14, 15 : Reserved
    - **Bit 16** : (*StatusErrorHardware*) : True if hardware error detected
    - Bits 17 to 31 : Reserved

6.3.2 Image Format

- **Sensor Width** (*SensorWidth*) : Get the physical width of the Sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r snsw**";
    Return by the sensor : Integer 16384.
  - ⇨ Can not be written;

- **Sensor Height** (*SensorHeight*) : Get the physical height of the Sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ No Access. Virtual command in xml"; Value always = 1

- **Width Max** (*WidthMax*) : Get the Maximum Width of the Sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ No Access. The value is mapped on "SensorWidth"

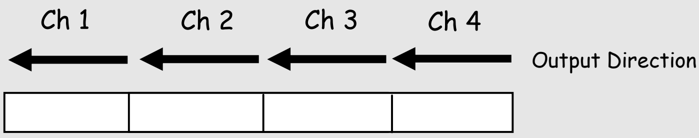
- **Height Max** (*HeigthMax*) : Get the Maximum height of the Sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ No Access. Virtual command in xml"; Value always = 1

- **Output mode** (*OutputMode*) : Set the CameraLink Output mode (refer also to Chap 3. : CameraLink Output Configuration). This command is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r mode**";
    Returned by the camera : Output mode from 0 to 3 (see table below).
  - ⇨ Write function : **"w mode"** <value> :
    detailed in the table below :

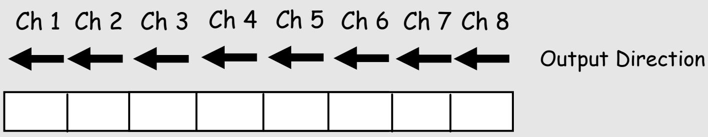
Modes	Connector CL1	Mode value
Medium 4 Outputs 8bits	4 x 85MHz 8 bits	0
Medium 4 Outputs 12bits	4 x 85MHz 12 bits	1
Full 8 Outputs 8bits	8 x 85MHz 8 bits	2
Full+ 10 Outputs 8bits (*)	10 x 85MHz 8 bits	3

Structure of the Camera Link Channels for interfacing

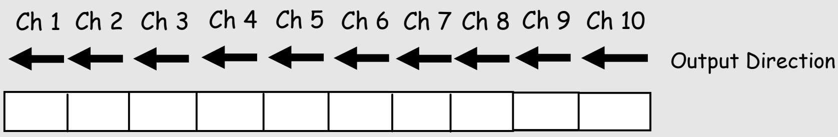
- **Medium Mode 4x4096 Pixels at 85MHz each Channel (4x2048 pixels in Binning Mode 1SB or 2SB)**
  4 Taps Separate, from Left to Right



- **FULL Mode 8x2048 Pixels at 85MHz each Channel (8x1024 pixels in Binning Mode 1SB or 2SB)**
  8 Taps Separate, from Left to Right



- **FULL+ Mode 10x1638 Pixels at 85MHz each Channel (10x819 pixels in Binning Mode 1SB or 2SB)**
  10 Taps Separate, from Left to Right : (*) Not available for EV71YC4MCL1605-**BA0** versions



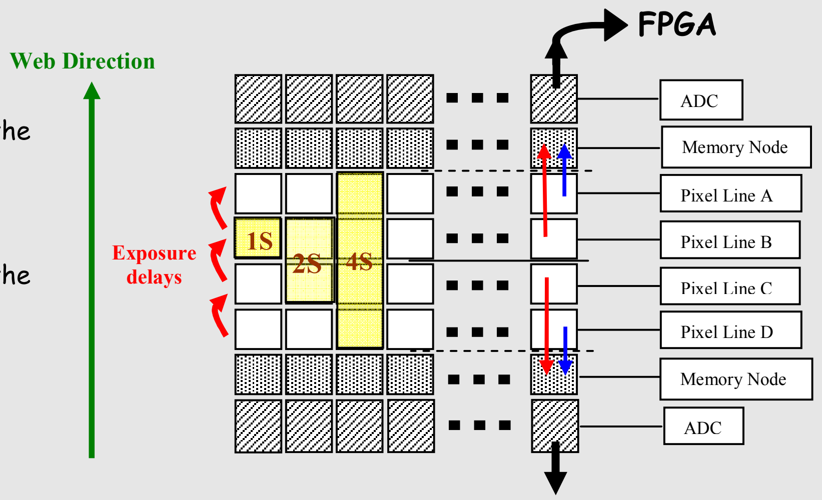
- **Output Frequency** (*OutputFrequency*) : Get the CameraLink Data Output Frequency. This value is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r clfq**";
    Return by the Camera : 0 (Frequency = 85MHz)
  - ⇨ Can not be written;

- **Sensor Mode** (*SensorMode*) : Defines the number of Line used on the Sensor. This command is available in the CommCam "Image Format Control" section :

Structure of the Sensor

In 2S Mode, the summation of the two lines is done in the FPGA :
B+C

In 4S Mode, the summation of the two double lines is done in the FPGA :
(AB)+ (BC)



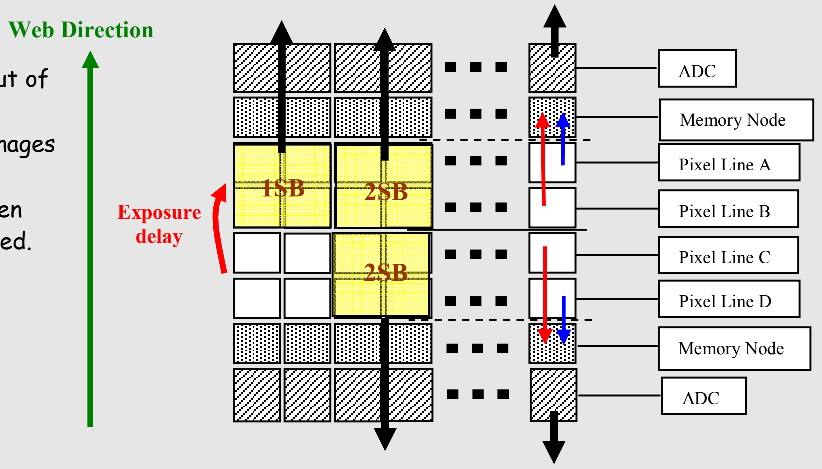
⇨ Read function : "**r smod**";
Returned by the camera : Integer from 0 to 4 (0 to 2 for BA0 version)

⇨ Write function : "**w smod**" <value> :
- "0" : "1S" mode or Single Line. Only one line (Line B) of the sensor is outputted.
- "1" : "2S" mode or Dual Lines. The two centred Lines B & C of the sensor are outputted. The Summation is done in the FPGA
- "2" : "4S" mode or Four Lines. All the lines of the sensor are outputted. A & B and B & C summation is done in the sensor. The overall summation is done in the FPGA

Only available with the EV71YC4MCL1605-**BA1** version :
- "3" : "1SB" mode : Binning mode (2x2) which outputs on line of 8k pixels in 10µmx10µm. The summation of the pixels is done in the sensor
- "4" : "2SB" mode : Binning mode 2 x (2x2) which outputs 2 lines of 8k pixels in 10µmx10µm. The summation of the pixels of each line is done in the sensor. The summation of the two lines is done in the FPGA. The appropriate delay for the exposure between the two lines of 8k pixels is managed automatically by the Sensor.

Binning modes (only available with EV71YC4MCL1605-**BA1** versions)

The two binning modes give an output of 8k pixels 10x10µm.
As for the 2S mode, the sensor manages the delay between the exposure necessary for a good acquisition when the double binning (2SB) mode is used.



Full Exposure Control

As the « 4S » mode is performing an internal Time delay exposure on the lines A & B and C & D, normally, the variation of the Exposure time should not possible in this sensor mode. Thanks to an e2v licensed solution, two of the Exposure controlled mode (Ext Trig with internal or External exposure control) are still available in 4S sensor TDE mode.

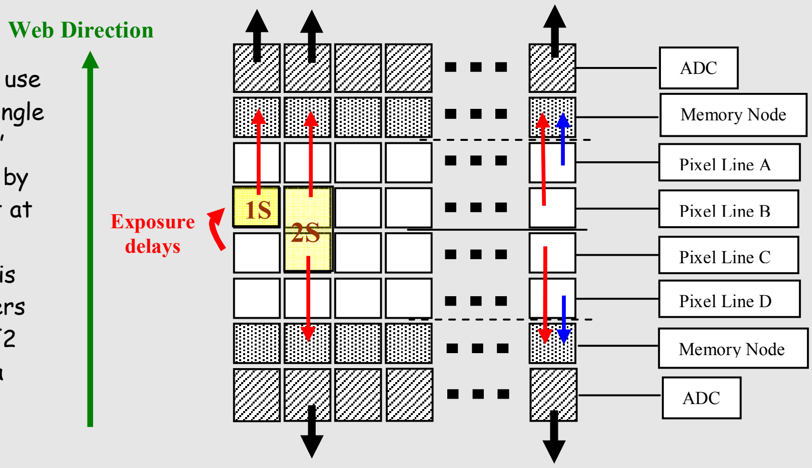
This is possible only with the BA1 version of the camera and with a firmware version starting at 2.0.4B. For all the other camera versions, the Exposure control is not possible with the 4S sensor mode.

- **Multi-Line**

- **Gain** (*MultiLineGain*) : Enables the MultiLine Gain of x0,5 . This value is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r mlig**";
    Return by the sensor : "0" if disabled (Gain x1 by default); "1" if Gain x0,5 activated.
  - ⇨ Write Function : "**w mlig** <value>"
    - "0" : Default Gain x1 is active.
    - "1" : Gain x0,5 is enabled (only when the 2S mode of the sensor is enabled)

Why Using a Multi-Line Gain of x0,5 ?

When the Light source is enough to use the "1S" mode of the Sensor (one single line), the best is to use 2 lines ("2S" mode) and then to divide the result by two by using the Multi-Line Gain set at "x0,5" :
In this case, the Full Well capacity is multiplied by x2 (two output registers are used) but the noise divided by √2 therefore the SNR is improved by a factor of √2.



- **Reverse Reading (X)** (*ReverseReading*) : Allows to output the line in the Reverse-X direction. This value is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r revr**";
    Return by the Camera : 0 or 1 (enabled/disabled)
  - ⇨ Write function : "**w revr** <value>";
    - "0" : Disabled.
    - "1" : Enables the reverse reading out (see below for "normal" direction)

- **Scan Direction** (*ScanDirection*) : Set the scan direction for the sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r scdi**";
    Return by the Camera : 0, 1 or 2 (Forward/reverse/external)
  - ⇨ Write function : "**w scdi** <value>";
    - "0" : Forward.
    - "1" : Reverse
    - "2" : Externally controlled (by CC3 of the CameraLink Sync signals)

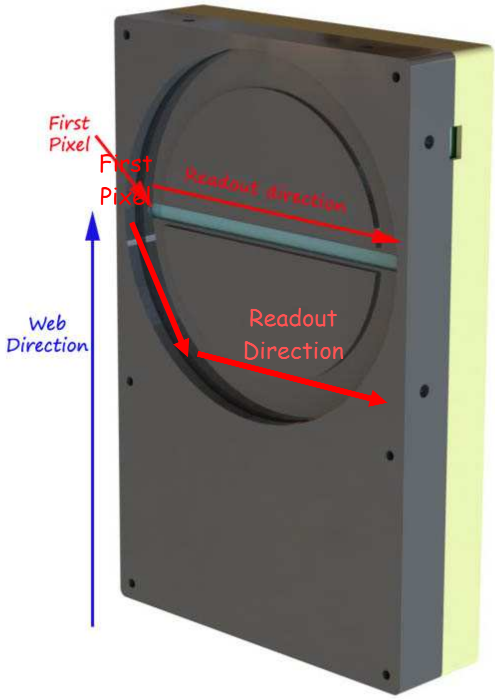
  Forward/reverse information has to be set correctly as soon as the Mode "2S" or "4S" of the sensor is set : In these tow modes, the sensor/Camera need to know what is the real order of the lines for the exposure delays.

The Forward direction is defined as detailed beside :

<u>Note</u> : : The minimum delay for the Camera to take in account a change in the ScanDirection value is :

- Using CC3 signal : **120ms**.
- Using serial command[(*)] : **180ms**

[(*)] After reception of the Command on the camera side



If the Camera is in **4S** Sensor mode, after changing of the scanning direction, the 5 first following triggers will be ignored in order to reinitialize the "Full Exposure Control" mode. Then the 3 following lines acquired will be more or less black because in 4S, 4 lines are required for a complete exposure.

In **2S or 2SB** Sensor modes, no Trigger will be lost after the change of scanning direction but the first line acquired will be more or less black as in 2S, 2 lines are required for a complete exposure.

In **1S** or **1SB** modes, nothing is lost an all lines received after the delay are correct.

This positioning takes also in account that the mode "Reverse X" is "Off" (Normal readout direction)

- **Test Image Selector** (*TestImageSelector*) : Defines if the data comes from the Sensor or the FPGA (test Pattern). This command is available in the CommCam "Image Format" section :
  - ⇨ Read function : "**r srce**";
    Returned by the camera : "0" if Source from the Sensor and "1 to 5" if test pattern active
  - ⇨ Write function : "**w srce**" <value> :
    - "0" : To switch to CCD sensor image
    - "1" : Grey Horizontal Ramp (Fixed) : **See AppendixA**
    - "2" : White Pattern (Uniform white image : 255 in 8Bits or 4095 in 12bits)
    - "3" : Grey Pattern (Uniform middle Grey : 128 in 8bits or 2048 in 12 bits)
    - "4" : Black Pattern (Uniform black : 0 in both 8 and 12 bits)
    - "5" : Grey vertical Ramp (moving)

  The test pattern is generated in the FPGA : It's used to point out any interface problem with the Frame Grabber.
  When any of the Test pattern is enabled, the whole processing chain of the FPGA is disabled.

6.3.3 Acquisition Control

This section deals with all the Exposure, Line period and synchronisation modes

- **Synchronisation Mode** (TriggerPreset) : Timed or Triggered, it defines how the grabbing is synchronized. This command is available in the CommCam "Acquisition Control" section :
  - ⇨ Read function : "**r sync**";
    Returned by the camera :
    - "**0**" : Internal Line Trigger with Exposure time Internally Controlled (Free Run). *Not available when Sensor mode is set in "4S" (whatever the firmware version)*
    - "**1**" : External Trigger with Exposure Time Internally Controlled. *Available when Sensor mode is set in "4S" only on BA1 with a firmware at least 2.0.4B*
    - "**2**" : External Trigger with maximum Exposure time
    - "**3**" : One External with Exposure Time Externally Controlled. The same Trigger signal defines the line period and its low level defines the exposure time. *Available when Sensor mode is set in "4S" only on BA1 with a firmware at least 2.0.4B*
    - "**4**" : Two External Triggers with Exposure Time Externally Controlled : CC2 defines the start of the exposure (and also the start Line) and CC1 defines the Stop of the exposure. *Not available when Sensor mode is set in "4S"*
    - "**5**" : Internal Line Trigger with maximum Exposure Time
  - ⇨ Write function : "**w sync**" <value>

> *The Timing diagrams associated to each Synchronization mode and the Timing values associated are detailed in the APPENDIX B of this document.*

- **Exposure time** (*ExposureTime*): Defines the exposure time when set in the Camera. This command is available in the CommCam "Acquisition Control" section :
  - ⇨ Read function : "**r tint**";
    Returned by the camera : Integer from 15 to 65535 (=1,5µs to 6553,5µs by step o 0,1µs)
  - ⇨ Write function : "**w tint**" <value> ;

    This value of exposure time is taken in account only when the synchronisation mode is "free run" (0) or "Ext Trig with Exposure time set" (1). Otherwise it's ignored.

> Due to the limitation of the timing pixel inside the sensor, the Exposure time has to be set by taking in account the limitation detailed in the APPENDIX B of this document.
> The **Minimum exposure time** which can be set is :
> - **4,2µs** with the EV71YC4MCL1605-BA0 version
> - **1,5µs** with the EV71YC4MCL1605-BA1 version

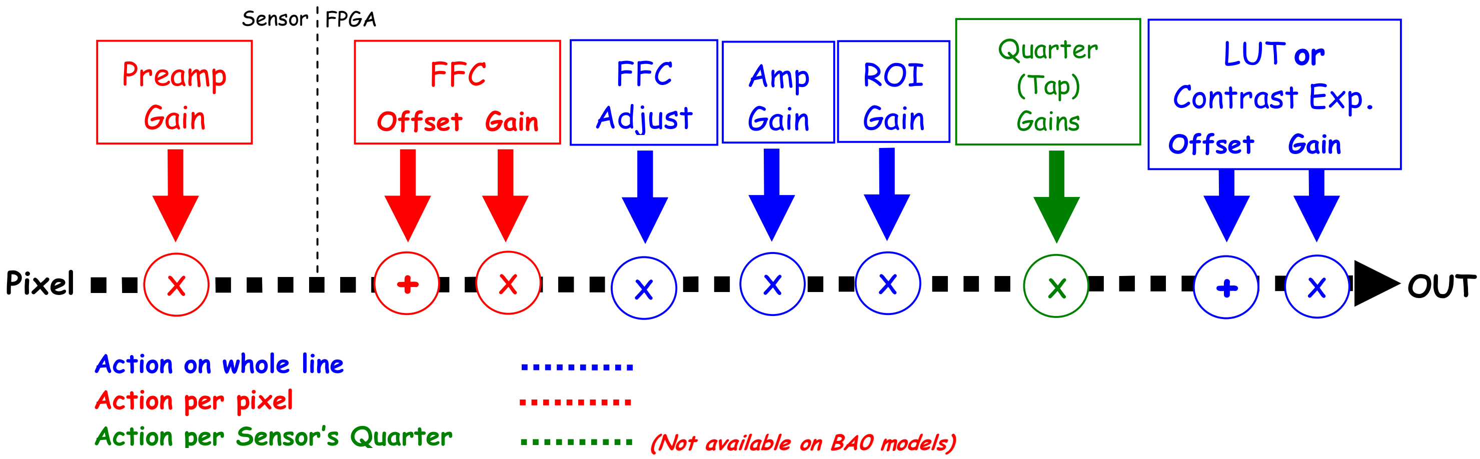
- **Line Period** (*LinePeriod*) : Defines the Line Period of the Camera in Timed mode. This command is available in the CommCam "Acquisition Control" section :
  - ⇨ Read function : "**r tper**";
    Returned by the camera : Integer from 1 to 65536 (=0,1µs to 6553,6µs by step o 100ns)
  - ⇨ Write function : "**w tper**" <value> ;

    The line period is active only in Free Run modes. It's also disabled if in this mode, the Integration time is set higher than the Line Period.

Minimum Line Period

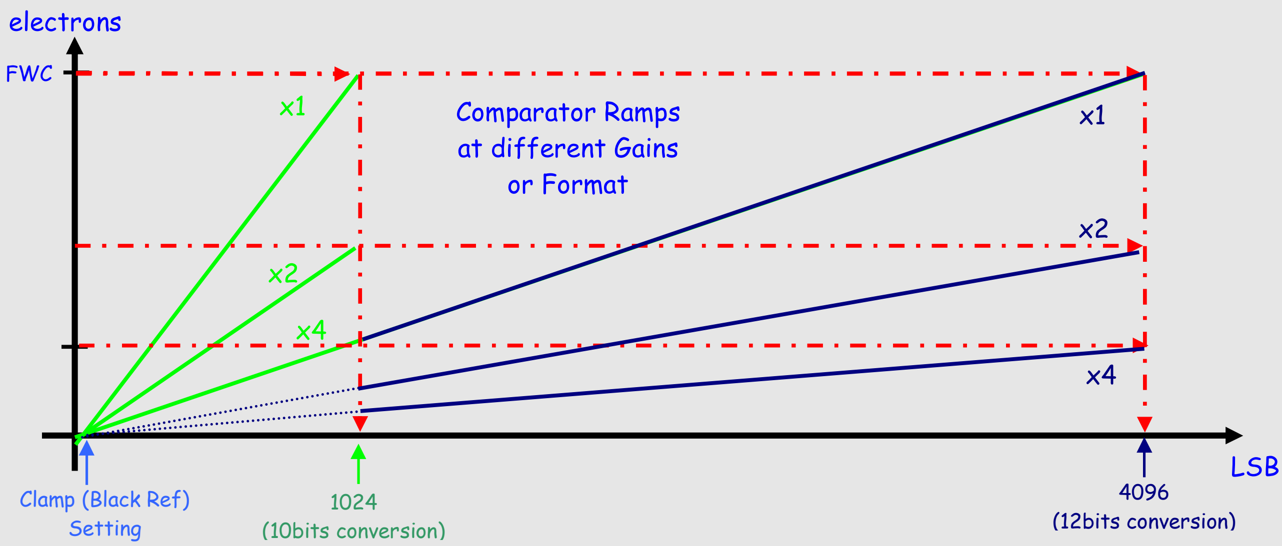
Medium modes (4 Taps) : **50µs**
Full Mode (8 Taps) : **25µs**
Full+ Mode (10 Taps) : **20µs** (not available with EV71YC4MCL1605-**BA0** version)

6.3.4 Gain and Offset



Analog Gain in the ADC

The only analog Gain available in the ELIIXA+ is located at the sensor level, in the ADC converter.
This "Preamp Gain" is in fact a variation of the ramp of the comparator of the ADC.
Then 3 Values are available : x1, x2 and x4. A gain x1 in a 12 bits conversion is equivalent to x4 in 10 bits.



- **Preamp Gain** : (*Gain* with *GainSelector= AnalogAll*)
  Set the Pre-amplification Gain. This command is available in the CommCam "Gain & Offset" section.
  ⇨ Read function : "**r pamp**";
  Returned by the camera : Integer corresponding to one of the 3 different step values :
    - 0 : x1 (0dB)
    - 1 : x2 (6dB)
    - 2 : x4 (12dB)

  ⇨ Write function : "**w pamp**" <int> ;

- **Gain:** (*Gain* with *GainSelector= GainAll*)
  Set the Amplification Gain. This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r gain**";
    Returned by the camera : Value from 0 to 6193 corresponding to a Gain range of 0dB to +8dB calculated as following : Gain(dB) = 20.log(1+ Gain/4096).
  - ⇨ Write function : "**w gain**" <int> ;

- **Tap Gain (*Gain* with *GainSelector*= *Tap*X) :** ***Not available on BA0 models***
  - ⇨ Read function : "r **fga**<tap>"; <tap> is 1 to 4
    Returns the Gain value for the tap. Ex : "*r fga1*" returns Gain value Tap1.
  - ⇨ Write function : "w fga<tap> <value>"
    - **<tap>** : 1 to 4
    - **<value> :** from -128 to +127 by step of 1 (0,0021dB each step)

- **Digital Gain** (*Gain* with *GainSelector=DigitalAll*) : Set the global Digital Gain. This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r gdig**";
    Returned by the camera : Integer value from 0 to 255. The corresponding Gain is calculated as 20log(1+val/64) in dB
  - ⇨ Write function : "**w gdig**" <int> ;

- **Digital Offset** (*BlackLevelRaw* with *BlackLevelSelector=All*) : Set the global Digital Offset. This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r offs**";
    Returned by the camera : Value from –4096 to +4095 in LSB
  - ⇨ Write function : "**w offs**" <int> ;

> *The Contrast Expansion (both Digital Gain & Offset) will be automatically disabled if the LUT is enabled..*

- **Tap Balance Gains Enable Switch (*TapBalanceGainEnable*) :** ***Not available on BA0 models***
  - ⇨ Read function : "r **tbe**";
    Returns the Gain value for the tap. Ex : "*r fga1*" returns Gain value Tap1.
  - ⇨ Write function : "w **tbe** <val>" with <val> : 0 or 1
    - 0 : Disables the Tap Balance Gains
    - 1 : Enables the Tap Balance Gains

- **ROI Gain () :** Set the Gain for the ROI Gain feature. ***Not available on BA0 models***
  - ⇨ Read function : "r **roig**";
    Returned by the camera : Value from 0 to 2047 (U1.9) corresponding to a Gain range from x1 to x1,999 and calculated as following : (1+ Gain/1024).
  - ⇨ Write function : "w **roig**" <value> ;

- **ROI Set ()** : Set the ROI and apply the Gain for ROI Gain Feature. *Not available on BA0 models*
  - ⇨ Read function : "r **rois**";
    Returns the ROI set for the last ROI gain command
  - ⇨ Write function : "w **rois** <val>" with <val> : Hexadecimal combination of Start and Stop address for the ROI (both on 16bits) : *0xStartAdr0xStopAdr*
    - Start address : from 0 to 16382 (0x000 to 0x3FFE)
    - Stop address : from 1 to 16383 (0x001 to 0x3FFF)

ROI Gain : How does it works

The ROI Gain feature comes in addition with the FFC (it's applied and calculated after).
The maximum complementary Gain ix x2.
It can be applied in 2 commands :

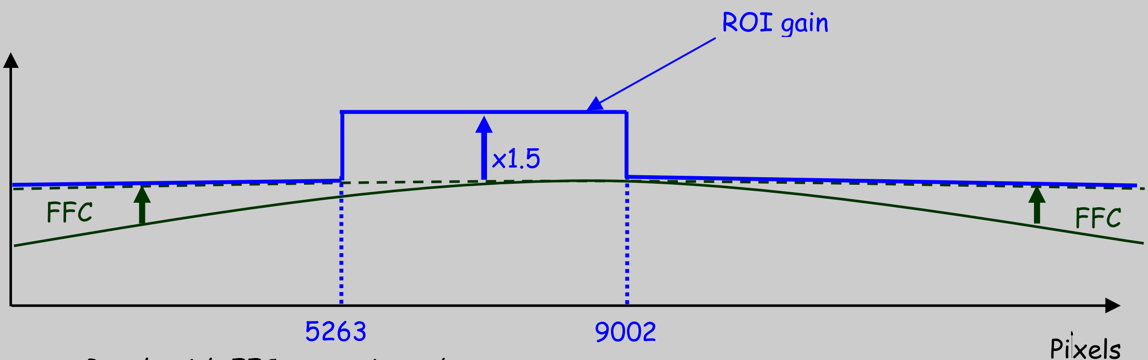
- *First, set the ROI Gain value.*
- *Second, set the ROI (Region of Interest).*

⇨ *This second command applies the Gain on the ROI in memory and this is immediately activated.*
⇨ *The ROI Gain is a "live" feature that can be overlapped but can't be saved in memory.*

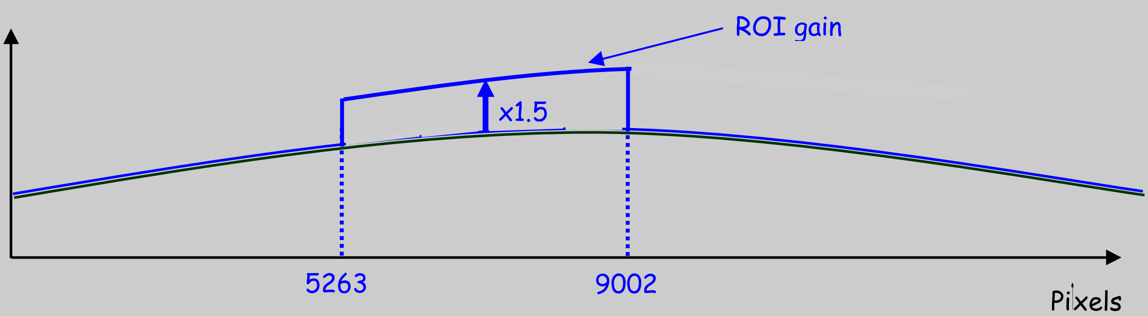
*Here is an example to apply a complementary gain of x1,5 (**512**) between the pixels #5263 (0x**148F**) and #9002 (0x**232A**), pixels included. The two commands are :*

- ***"w roig 512"***
- ***"w rois 0x148F232A"***

Result with FFC activated :



Result with FFC not activated :



6.3.5 Flat Field Correction

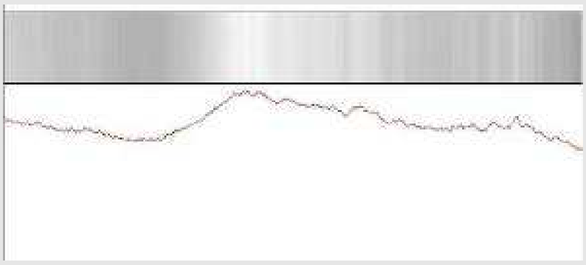
How is performed the Flat Field Correction ?

What is the Flat Field correction (FFC) ?

The Flat Field Correction is a digital correction on each pixel which allows :

- To correct the Pixel PRNU (Pixel Response Non Uniformity) and DSNU (Dark Signal Non Uniformity)
- To Correct the shading due to the lens
- To correct the Light source non uniformity

Before



After



How is calculated / Applied the FFC ?

The FFC is a digital correction on the pixel level for both Gain and Offset.

- Each Pixel is corrected with :
  - An Offset on 8 bits (Signed Int 5.3). They cover a dynamic of ±16LSB in 12bits with a resolution of 1/8 LSB 12bits.
  - A Gain on 14 bits (Unsigned Int 14) with a max gain value of x5(*)

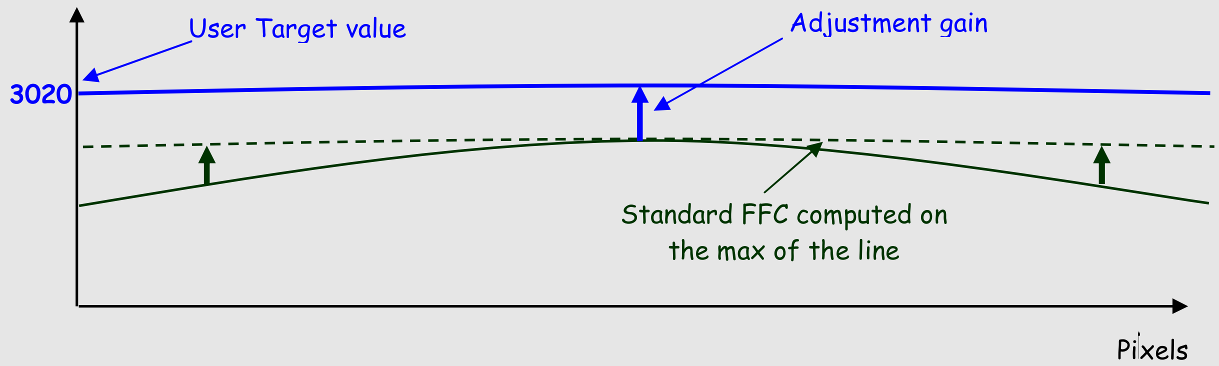
The calculation of the new pixel value is : **P' = (P + Off).(1 + Gain/1024(*))**

The FFC processing can be completed with an automatic adjustment to a global target. This function is designed as "**FFC Adjust**". This adjustment to a User target is done by an internal hidden gain which is re-calculated each time the FFC is processed while the FFC adjust function is enabled.

The FFC is always processed with the max pixel value of the line as reference. If enabled, the FFC adjust module (located at the output of the FFC module) calculates the adjustment gain to reach the target defined by the User.

When the FFC result is saved in memory, the adjust gain and target are saved in the same time in order to associate this gain value with the FFC result.

*(*) : Up to the firmware version 2.0.5A, the Gain resolution was : **1 + Gain/8192** with a range limited at **x3***



How to perform the Flat Field Correction ?

FPN/DSNU Calibration

- Cover the lens
- Launch the FPN Calibration : Grab and calculation is performed in few seconds

PRNU Calibration

The User must propose a white/gray uniform target to the Camera (not a fixed paper).
The Gain/Light conditions must give a non saturated image in any Line.
The Camera must be set in the final conditions of Light/ Gain and in the final position in the System.
I f required, set a user target for the FFC adjust and enable it.

- White uniform (moving) target
- Launch the FFC
- Enable the FFC
- You can save the FFC result (both FPN+PRNU in the same time) in one of the 4 x FFC User Banks.
- The user target and Gain are saved with the associated FFC in the same memory.

Advices

The ELIIXA+ Cameras have 4 x FFC Banks to save 4 x different FFC calibrations. You can use this feature if your system needs some different conditions of lightning and/or Gain because of the inspection of different objects : You can perform one FFC per condition of Gain/setting of the Camera (4 Max) and recall one of the four global settings (Camera Configuration + FFC + Line Balance) when required.

6.3.5.1 Activation and Auto-Adjust

- **FFC Activation** (*FFCEnable*) : Enable/disable the Flat Field Correction. This command is available in the CommCam "Flat Field Correction" section :
  ⇨ Read function : "**r ffcp**" : Returns the FFC Status (0 if disabled, 1 if enabled)
  ⇨ Write function :
    - "**w ffcp 1**" : Enable the FFC.
    - "**w ffcp 0**" : Disabled the FFC

- **FFC Adjust Function** : This Feature is available in the CommCam "Flat Field Correction/ Automatic Calibration" section :
  - **Gains adjust** (*FFCAdjust*)**:** Enable/Disable the function
    ⇨ Read function : "**r ffad**". Returns the status of the function (0 if disabled)
    ⇨ Write function :
      - "**w ffad 0**" : Disable the FFC Adjust function.
      - "**w ffad 1**" : Enable the FFC Adjust function.

  - **Auto Adjust Target Level** (*FFCAutoTargetLevel*)**:** set the value for the User Target.
    ⇨ Read function : "**r tfad**". Returns the Target value (from 0 to 4095)
    ⇨ Write function : "**w tfad <value>**" : Set the Target Value (in 12bits)
    ⇨

<u>FFC Adjust : A good usage.</u>

When there are several Cameras to set up in a system on a single line, the most difficult is to have a uniform lightning whole along the line.
If each Camera performs its own Flat field correction, relative to the max of each pixel line, the result will be a succession of Camera lines at different levels.
=> The FFC Adjust function allows to set the same target value for all the Cameras in the system and then to get a perfect uniform line whole along the system with a precision of 1 LSB to the Target.
The Maximum correction is x2 the highest value of the line.

The reasonable value for the User Target is not more than around 20% of the max value of the line.

6.3.5.2 Automatic Calibration

- **FPN/DSNU Calibration :**

  - **FPN Calibration Control** (*FPNCalibrationCtrl*) : Launch or abort of the FPN process for the Offsets calculation. These commands are available in the CommCam "Flat Field Correction / Automatic Calibration " section :
    - ⇨ Read function : "**r calo**" : Returns the FPN Calculation Process Status (0 if finished, 1 if processing)
    - ⇨ Write function :
      - "**w calo 1**" : Launch the FPN Calibration Process.
      - "**w calo 0**" : Abort the FPN Calibration Process.

  - **FPN Coefficient Reset** (*FPNReset*) : Reset the FPN (Offsets) coefficient in Memory. This command is available in the CommCam "Flat Field Correction / Manual Calibration " section :

    - ⇨ Write function : "**w rsto 0**" : Reset (set to 0) the FPN coefficients in memory. This doesn't affect the FFC User Memory Bank but only the active coefficients in Memory.

- **PRNU Calibration :**
  - **PRNU Calibration Control** (*FFCCalibrationCtrl*) : Launch or abort of the PRNU process for the Gains calculation. This command is available in the CommCam "Flat Field Correction / Automatic Calibration " section :
    - ⇨ Read function : "**r calg**" : Returns the PRNU Calculation Process Status (0 if finished, 1 if processing)
    - ⇨ Write function :
      - "**w calg 1**" : Launch the PRNU Calibration Process.
      - "**w calg 0**" : Abort the PRNU Calibration Process.

  - **PRNU coefficient Reset** (*PRNUReset*) : Reset the PRNU (Gains) coefficient in Memory. This command is available in the CommCam "Flat Field Correction / Manual Calibration " section :
    - ⇨ Write function : "**w rstg 0**" : Reset (set to "x1") the PRNU coefficients in memory. This doesn't affect the FFC User Memory Bank but only the active coefficients in Memory.

Some Warnings can be issued from the PRNU/FPN Calibration Process as "pixel Overflow" of "Pixel Underflow" because some pixels have been detected as too high or too low in the source image to be corrected efficiently.
The Calculation result will be proposed anyway as it's just a warning message.
The Status Register is the changed and displayed in CommCam "Status" section :



Register status is detailed chap §6.3.3.

6.3.5.3 Manual Flat Field Correction

The FFC Coefficients can also be processed outside of the Camera or changed manually by accessing directly their values in the Camera : This is the "Manual" FFC.

In CommCam, the User can access to a specific interface by clicking on "click for extended control" in both "Manual FFC calibration" and "Manual FPN calibration sections" :





This will allow the user to upload/download out/in the Camera the FFC coefficients in/from a binary or text file that can be processed externally.

> It is recommended to setup the baud rate at the maximum value possible (115000 for example) otherwise the transfer can take a long time.

- **FPN coefficients modification :** Direct access to the FPN coefficients for reading or writing.
  The FPN coefficients are read packets of x128 coefficients :

  - ⇨ Read function : "**r ffco <addr>**" : Read 128 consecutive FPN user coefficients starting from **<addr>** address. Returned value is in hexadecimal, without space between values (one unsigned short per coefficient).
  - ⇨ Write function :" **w ffco <addr><val>** : Write 128 consecutive FPN user coefficients starting from the **<addr>** address. **<val>** is the concatenation of individual FPN values, without space between the values (one unsigned short per coefficient).

- **PRNU coefficients modification :** Direct access to the PRNU coefficients for reading or writing.
  The PRNU coefficients are read packets of x128 coefficients :

  - ⇨ Read function : "**r ffcg <addr>**" : Read 128 consecutive PRNU user coefficients starting from **<addr>** address. Returned value is in hexadecimal, without space between values (one unsigned short per coefficient).
  - ⇨ Write function :" **w ffcg <addr><val>** : Write 128 consecutive PRNU user coefficients starting from the **<addr>** address. **<val>** is the concatenation of individual PRNU values, without space between the values (one unsigned short per coefficient).

6.3.5.4 FFC User Bank Management

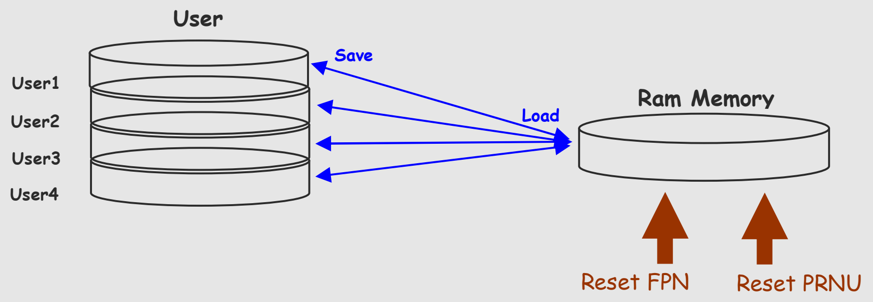
The new-processed FFC values can be saved or restored in/from 4 x User banks.
Both Gains and Offsets in the same time but also the FFC Adjust User target and associated gain.
These functions are available in the Flat Field correction/Save & Restore FFC section :

- **Restore FFC from Bank** (*RestoreFFCFromBank*) : Restore the FFC from a Bank in the current FFC.
  ⇨ Read function : "**r rffc**" : Get the current FFC Bank used
  Returned by the camera : 0 for Factory bank or 1 to 4 for User banks
  ⇨ Write function : "**w rffc <val>**" : Bank <val> 1 to 4 for User banks

  Note : Factory means neutral FFC (no correction).

- **Save FFC in User Bank** (*SaveFFCToBank*) : Save current FFC in User Bank
  ⇨ Can not de read
  ⇨ Write function : "**w sffc <val>**" : User bank <val> if from 1 to 4.

FFC User Bank Usage



At the power up :
- Last User Bank used is loaded in RAM

Reset a User bank :
- Reset the RAM (FPN/PRNU individually)
- Save in the bank to reset

6.3.6 Look Up Table

The User can define an upload a LUT in the Camera that can be used at the end of the processing.
The LUT is defined as a correspondence between each of the 4096 gray levels (in 12 bits) with another outputted value. For example, a "negative" or "reverse" LUT is the following equivalence :

Real value	Output value
0	4095
1	4094
2	4093
…	

Then the size of each value is 12bits but the exchanges with the Application/PC are done on 16 bits :
For 4096 gray levels (from 0 to 4095) the total file size for a LUT is 8Ko.

If this LUT is enables, the "Contrast Expansion" feature (digital Gain and Offset) will be disabled

- **LUT Enable** (*LUTEnable*) : Enable the LUT and sizable the Digital Gain / Offset
  This function is available in the LUT section :.
  - ⇨ Read function : "**r lute**" : Get the LUT status
    Returned by the camera : 0 is LUT disabled, 1 if enabled
  - ⇨ Write function : "**w lute <val>**" : <val> is 0 for disable, 1 for enable

- **Upload / Download the LUT coefficients :** Direct access to the LUT coefficients for reading or writing. In CommCam, the User can access to a specific interface by clicking on "click for extended control" in the LUT section :



- ⇨ Read function : "**r lutc <addr>**" : Read 128 LUT coefficients starting from address **<addr>** from 0 to 4095-128. Returned value is the concatenation in hexadecimal of individual LUT values, without space between values. (one unsigned short per coefficient)
- ⇨ Write function :" **w lutc <addr><val>** : Write 128 LUT coefficients starting from address **<addr>** form 0 to 4095-128. **<val>** is the concatenation in hexadecimal of individual LUT values, without space between values. (one unsigned short per coefficient)

- **Save & Restore LUT in User Banks** : The LUT loaded in RAM memory can be saved or restored in/from 4 User banks.

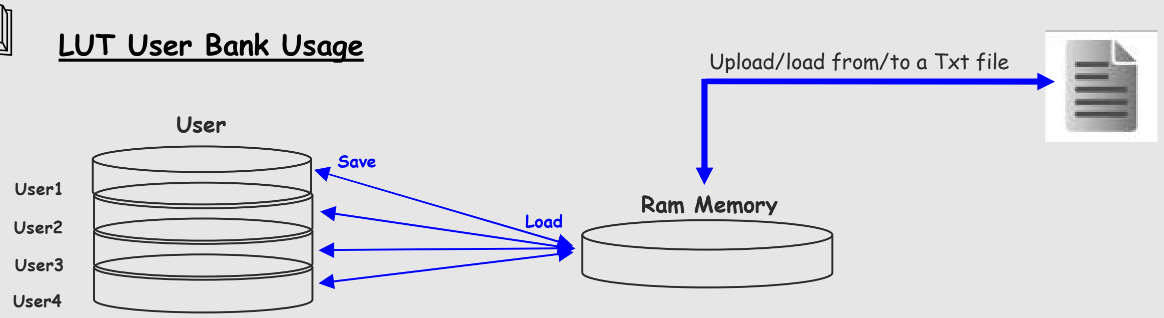
  These functions are available in the LUT/Save & Restore LUT Settings section :

  - o **Restore LUT from Bank** (*RestoreLUTFromBank*) : Restore the LUT from a User Bank in the current RAM Memory.
    - ⇨ Read function : "**r rlut**" : Get the current LUT Bank used
      Returned by the camera : 1 to 4 for User banks
    - ⇨ Write function : "**w rlut <val>**" : Bank <val> 1 to 4 for User banks

  - o **Save LUT in User Bank** (*SaveLUTToBank*) : Save current LUT in User Bank
    - ⇨ Can not de read
    - ⇨ Write function : "**w slut <val>**" : User bank <val> if from 1 to 4.

  The bank number <val> is given in (*LUTSetSelector*)

LUT User Bank Usage



At the power up :
- Last User Bank used is loaded in RAM

6.3.7 Statistics and Line Profile

This function allows the User to get some statistics on a pre-defined ROI. On request, the Camera acquires and then calculates some key values as the min, the max, the average or the standard deviation in this Region of Interest.
The grab and calculation command and also the collection of the results is not performed in real time as it is done through the serial connection.
This function and the results are available in CommCam in the "Line Profile Average" Section :

- **Line Profile average measurement** (*LineAverageProfile*) : Control the grab and computation of the statistics.
  ⇨ Read function : "**r pixs**" : Get the status of the calculation
  Returned by the camera : 0 : finished, 1: running
  ⇨ Write function :
  - "**w pixs 1**" : Start the accumulation and then the computing
  - "**w pixs 0**" : Abort the computing.

  The Calculated values are detailed as following :
  - **Pixel average Value** (*PixelROIMean*) : Average gray level value calculated on whole Region of interest
    ⇨ Read function : "**r pavr**" : Get the average value
    Returned by the camera : Unsigned format value : U12.4
  - **Pixel Standard deviation** (*PixelROIStandardDeviation*) : standard deviation of all the pixel gray level values of Region of interest
    ⇨ Read function : "**r pstd**" : Get the standard deviation
    Returned by the camera : Unsigned format value : U12.4
  - **Pixel Min value** (*PixelROIMin*) : Minimum gray level pixel value on the whole region of interest.
    ⇨ Read function : "**r pmin**" : Get the Minimum value
    Returned by the camera : Unsigned format value : U12.4
  - **Pixel Max Value** (*PixelROIMax*) : Maximum gray level pixel value on the whole region of interest
    ⇨ Read function : "**r pmax**" : Get the maximum value
    Returned by the camera : Unsigned format value : U12.4

- **Pixel access Line number** (*PixelAccessLineNumer*) : Set the number of lines to accumulate.
  ⇨ Read function : "**r pixl**" : Get the number of line
  Returned by the camera : 1, 256, 512 or 1024
  ⇨ Write function : "**w pixl <val>**" : Set the number of lines. <val> is 1, 256, 512 or 1024.

- **Pixel ROI Start** (*PixelRoiStart*) : Set the Region of Interest start position.
  ⇨ Read function : "**r prod**" : Get the starting pixel
  Returned by the camera : value between 0 and 16383
  ⇨ Write function : "**w prod <val>**" : Set the starting pixel. <val> is between 0 and 16383

- . **Pixel ROI Width** (*PixelRoiWidth*) : Set the Width of the Region of Interest.
  ⇨ Read function : "**r prow**" : Get the width in pixel
  Returned by the camera : value between 1 and 16384
  ⇨ Write function : "**w prow <val>**" : Set the ROI width in pixels. <val> is between 1 and 16384

After performing a line profile measurement, all the values computed which are described below are not refreshed automatically in CommCam : You have to right-click on each value and ask for an individual refresh.

6.3.8 Privilege Level

There are 3 privilege levels for the camera :

- Factory (0) : Reserved for the Factory
- Integrator (1) : Reserved for system integrators
- User (2) : For all Users.

The Cameras are delivered in Integrator mode. They can be locked in User mode and a specific password is required to switch back the Camera in Integrator mode. This password can be generated with a specific tool available from the hotline (hotline-cam@e2v.com)

This function is available in the Privilege section :

- **Privilege level Management** (*PrivilegeLevel*) : Get the current Camera privilege level..
  - ⇨ Read function : "**r lock**" : Get the current privilege
    Returned by the camera : 0 to 2
  - ⇨ Write function : "**w lock <val>**" : <val> is as follow
    - **2** : Lock the Camera in Integrator or "privilege User"
    - **<computed value>** : Unlock the Camera back in Integrator mode

6.3.9 Save & Restore Settings

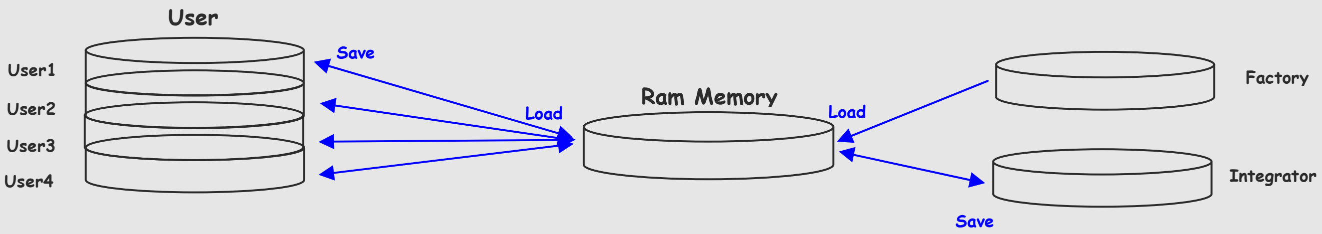
The settings (or Main configuration) of the Camera can be saved in 4 different User banks and one Integrator bank. This setting includes also the FFC and LUT enable
This function is available in the Save & Restore Settings section :

- **Load settings from Bank :** Allows to restore the Camera settings.
  - ⇨ Read function : "**r rcfg**" : Get the current Tap Bank in use
  - ⇨ Write function : "**w rcfg <val>**" : Load settings from bank <val> (0: Factory , 1 to 4 for Users, 5 for Integrator)

- **Save settings to Bank :** Allows to save the Camera settings in User or Integrator Bank
  - ⇨ Write function : "**w scfg <val>**" : Save the current settings in the User bank <val> (1 to 4 for User, 5 for Integrator)

> The integrator bank (User Set5) can be written only if the Camera is set in integrator mode (Privilege level = 1). This integrator bank can be used as a « Factory default » by a system integrator.

Configuration Bank Usage



At the power up : Last User Bank used is loaded in RAM
"Integrator" Bank (5) can be locked by switching the Camera in "User" mode (cf : Privilege feature).
Then it can't be saved any more without switching back the Camera in "Integrator" Mode.

7 APPENDIX A: Test Patterns

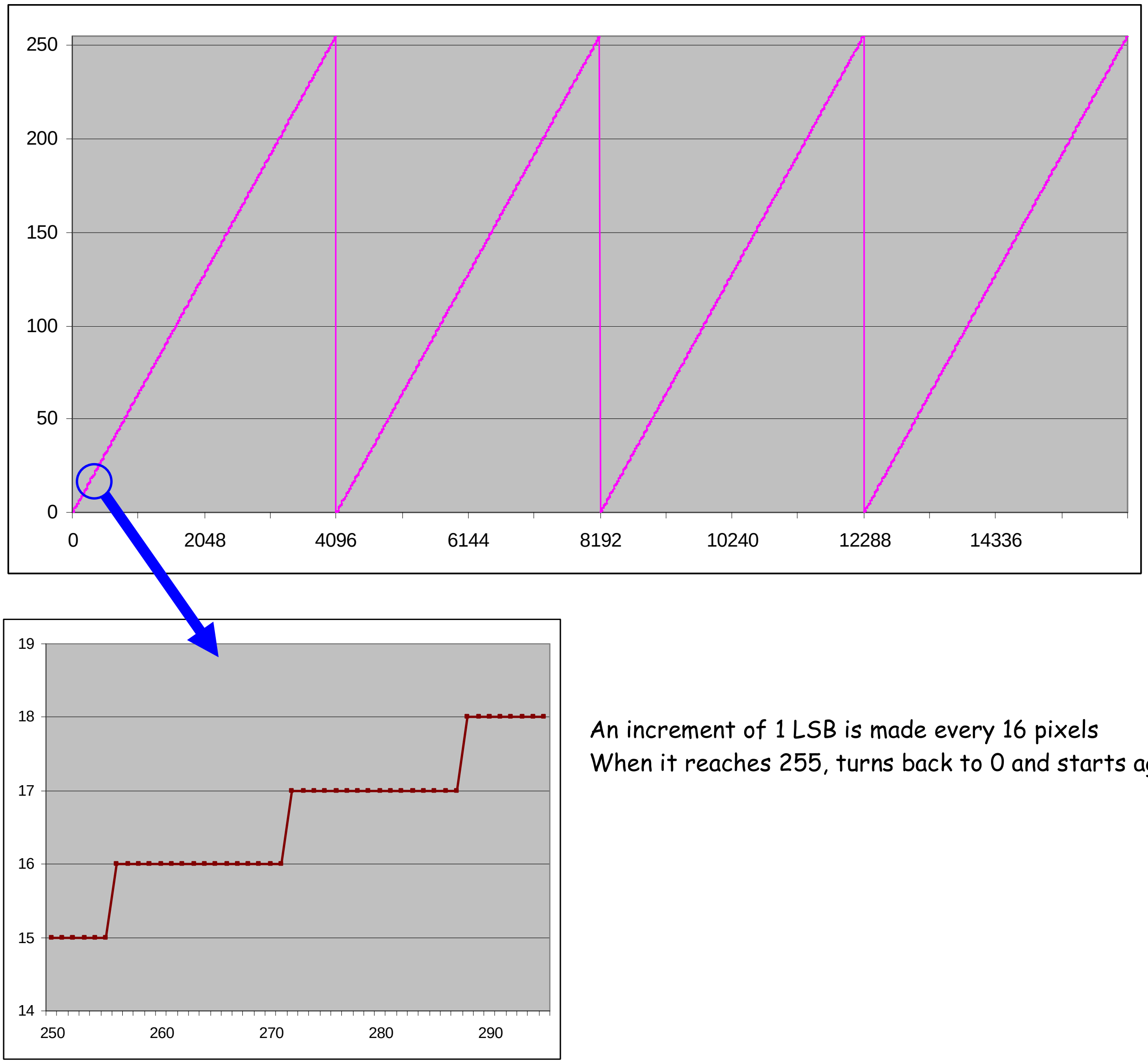
7.1 Test Pattern 1: Vertical wave

The Test pattern 1 is a vertical moving wave : each new line will increment of 1 gray level in regards with the previous one.

- In 12 bits the level reaches 4095 before switching down to 0
- In 8 bits the level reaches 255 before switching down to 0

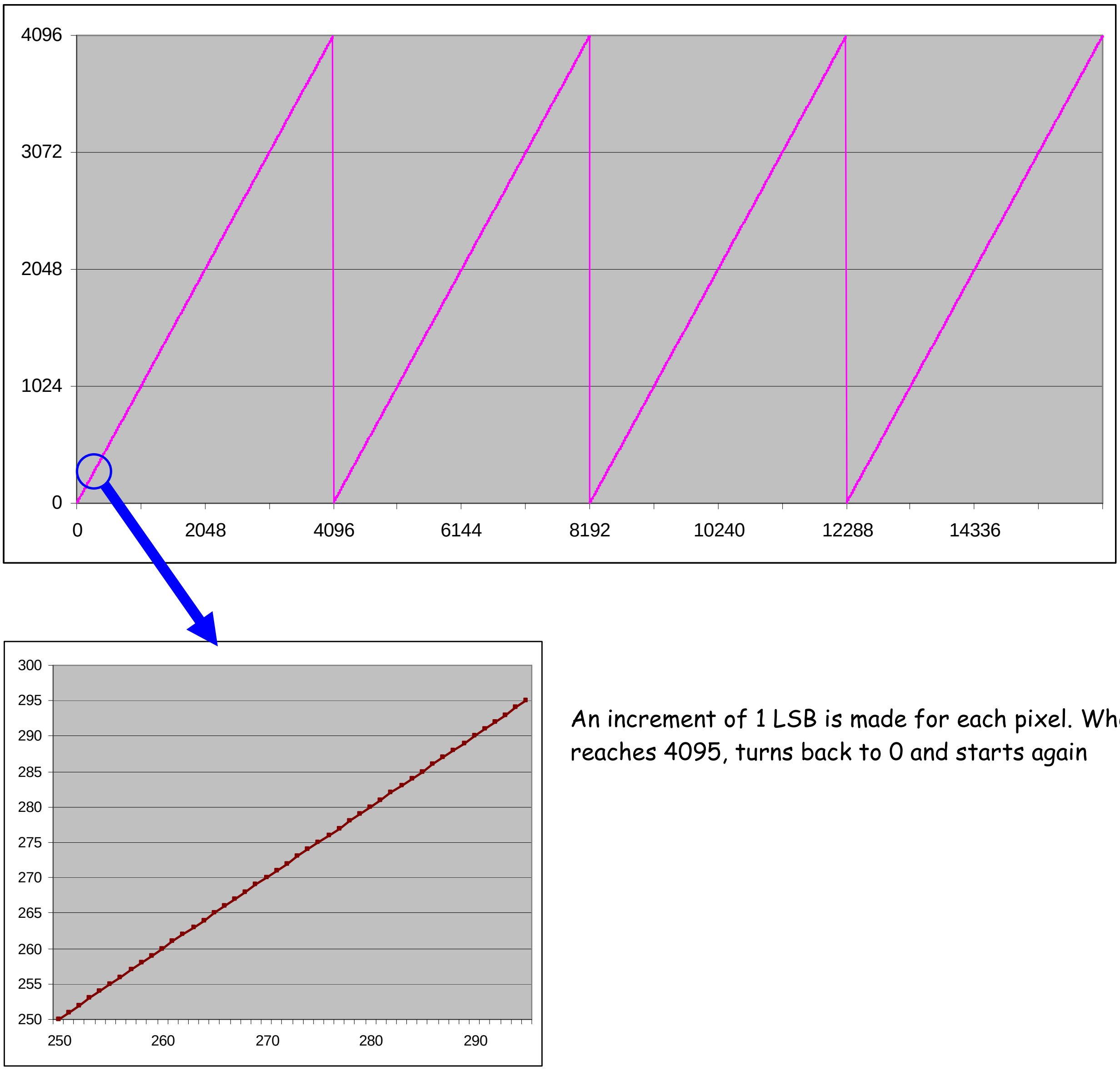
7.2 Test Pattern 2: Fixed Horizontal Ramps

7.2.1 In 8 bits (Full) format – No Binning (16384 pixels)



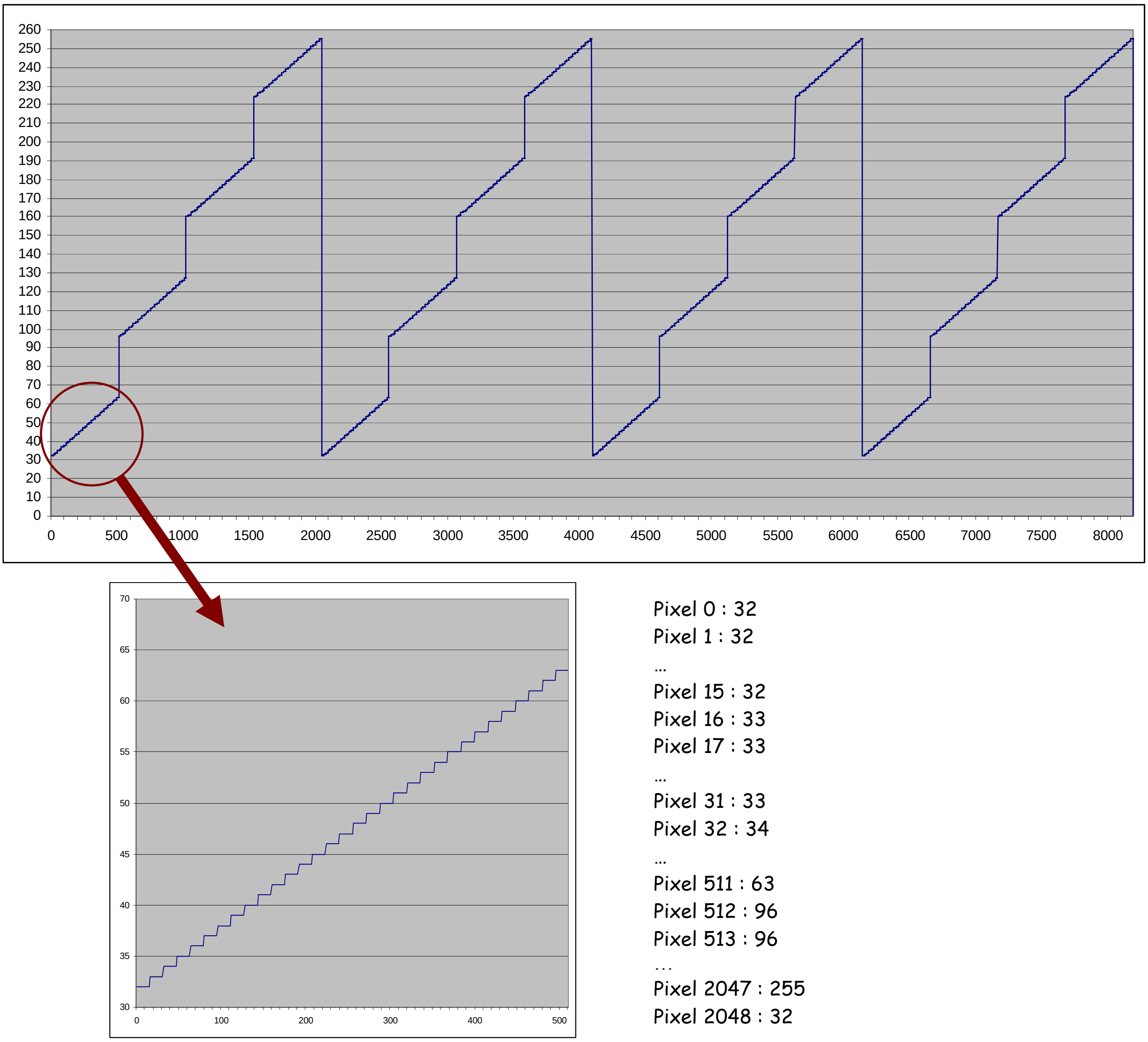
An increment of 1 LSB is made every 16 pixels
When it reaches 255, turns back to 0 and starts again

7.2.2 In 12 bits (Medium) format – No Binning (16384 pixels)



An increment of 1 LSB is made for each pixel. When it reaches 4095, turns back to 0 and starts again

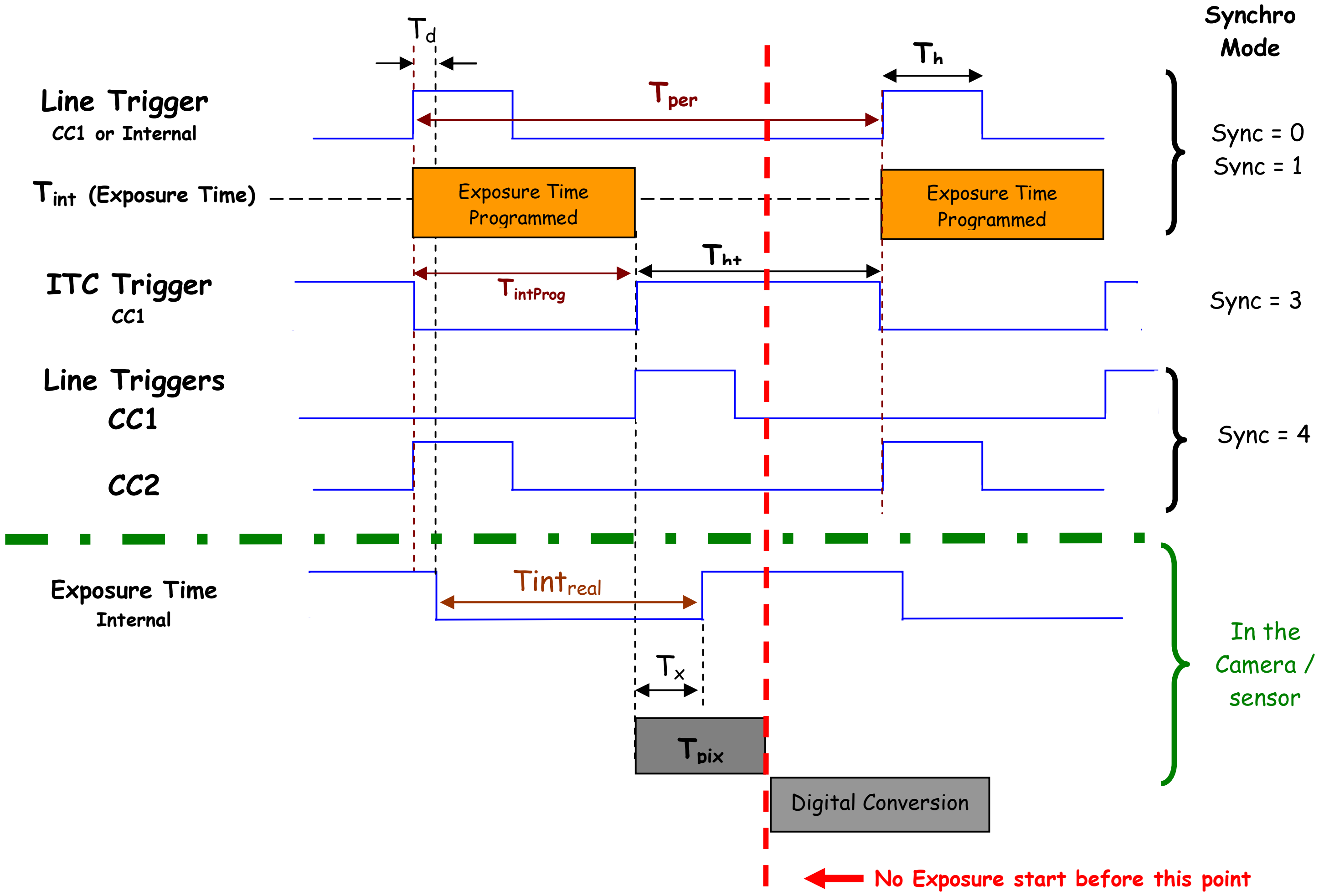
7.2.3 In 8/12 bits Full/Medium format with Binning (8192 Pixels)



Pixel 0 : 32
Pixel 1 : 32
…
Pixel 15 : 32
Pixel 16 : 33
Pixel 17 : 33
…
Pixel 31 : 33
Pixel 32 : 34
…
Pixel 511 : 63
Pixel 512 : 96
Pixel 513 : 96
…
Pixel 2047 : 255
Pixel 2048 : 32
…

8 APPENDIX B: Timing Diagrams

8.1 Synchronization Modes with Variable Exposure Time



T_{pix} : Timing Pixel. During this uncompressible period, the pixel and its black reference are read out to the Digital converter. During the first half of this timing pixel (read out of the black reference), we can consider that the exposure is still active.

Digital Conversion : During the conversion, the analog Gain is applied by the gradient of the counting ramp (see next chapter : Gain & Offset). The conversion time depends on the pixel format :

- 8 or 10 bits : **6µs**
- 12 bits : **24µs**

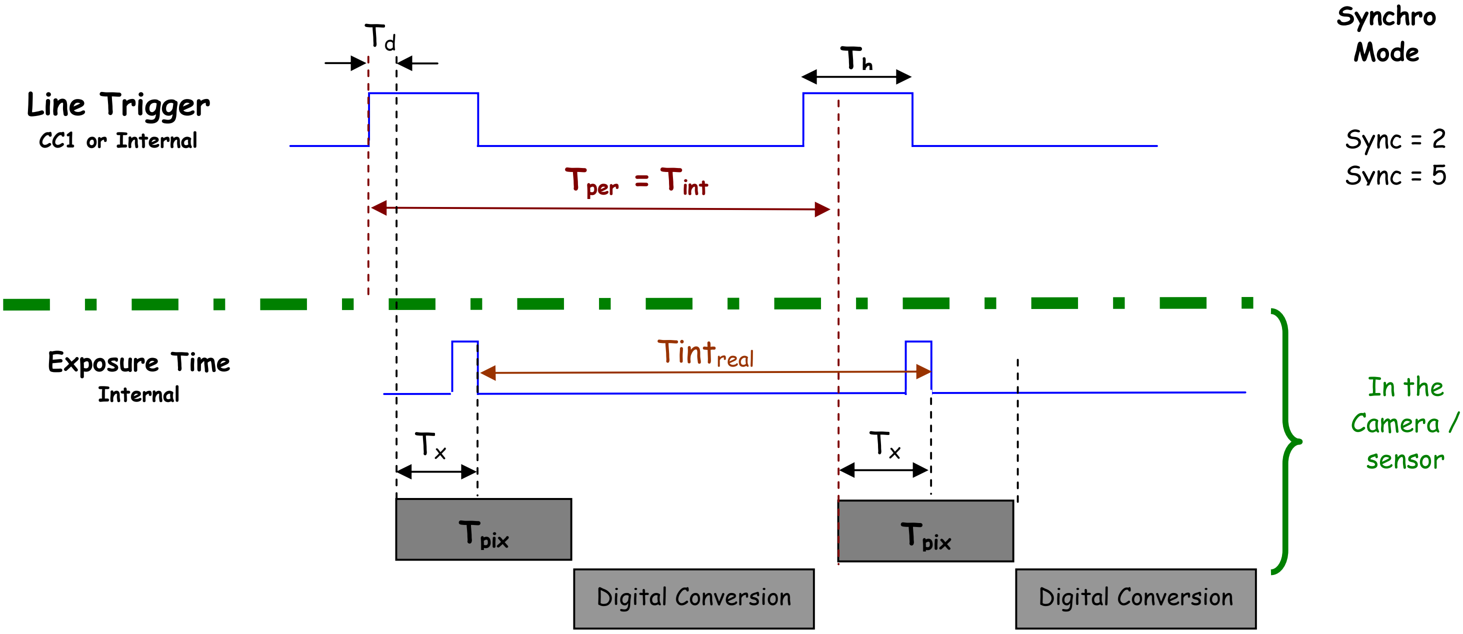
This conversion is done in masked time, eventually during the next exposure period.

T_d : Delay between the Start exposure required and the real start of the exposure.

If T_{per} is the Line Period (internal or external coming from the Trigger line), in order to respect this line Period, the Exposure Time as to be set by respecting : $T_{int} + T_{pix} <= T_{per}$
Then, the real exposure time is : $Tint_{real} = T_{int} + T_x - T_d$.
In the same way, The high level period of the Trig signal in sync=3 mode, $T_{ht} >= T_{pix}$

For a Line Period of *LinePer*, the maximum exposure time possible without reduction of line rate is : $Tint_{max} = T_{per} - T_{pix}$ (T_{pix} is defined above) but the effective Exposure Time will be about $Tint_{real} = T_{int} + T_x - T_d$.

8.2 Synchronisation Modes with Maximum Exposure Time

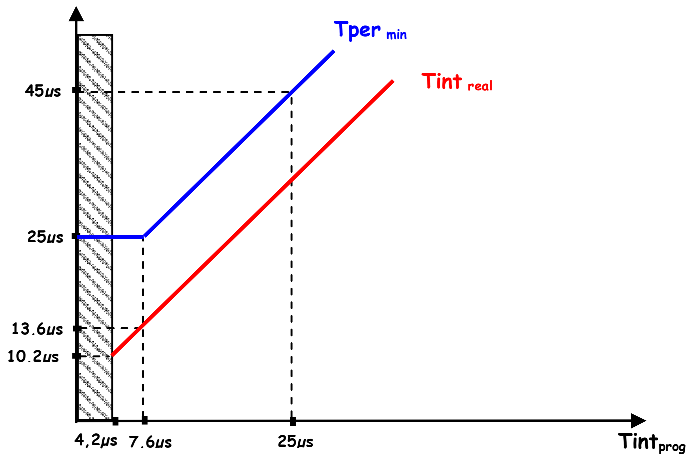


In these modes, the rising edge of the Trigger (internal or External) starts the readout process (T_{pix}) of the previous integration. The Real exposure time ($Tint_{real}$) is finally equal to the Line Period (T_{per}) even if it's delayed from ($T_x + T_d$) from the rising edge of the incoming Line Trigger.

8.3 Timing Values

8.3.1 EV71YC4MCL1605-BA0 version

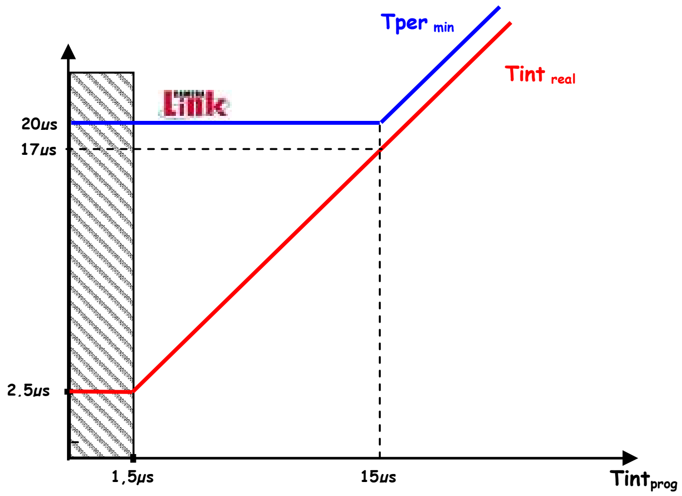
Label	Min	Unit
T_{pix}	17,4	µs
T_x	9.7	µs
T_h	0,120	µs
T_{ht}	T_{pix}	µsec
$T_{IntProg}$	4,2	µs
T_d	3.69	µs



Due to the high value of the Timing pixel in this EV71YC4MCL1605-BA0 version and the associated limitations, we recommend the use of synchro modes 2 and 5 (maximum exposure time) and then to pulse the light for a exposure control, specially for the highest line rates.

8.3.2 EV71YC4MCL1605-BA1 version

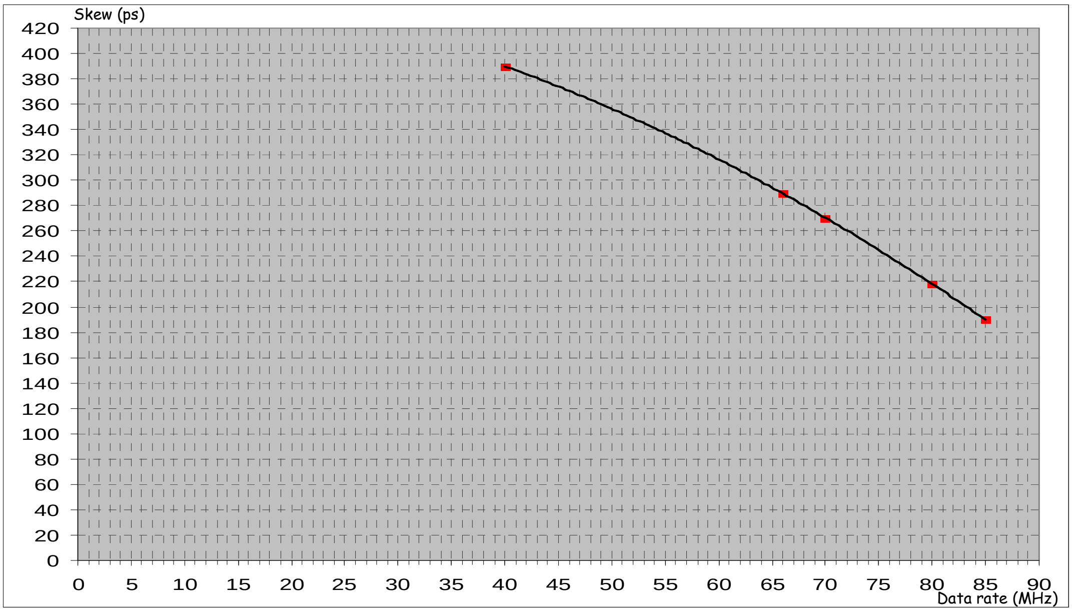
Label	Min	Unit
T_{pix}	5	µs
T_x	3,1	µs
T_h	0,120	µs
T_{ht}	T_{pix}	µsec
T_d	1,1	µs



9 APPENDIX C: CameraLink Data Cables

You may check the compliance of your CameraLink cables with the transportation of the 85MHz data rate.
The main parameter to be checked in the cable specification is the skew (in picoseconds)
This parameter is given for a dedicated maximum value per meter of cable (as max : 50ps/m)

The CameraLink Standards defines the maximum total skew possible for each data rate :



Here is a following example of cable and the cable length limitation in accordance with the standard :

Conductor Size:	28 AWG Stranded
Propogation Velocity:	1.25 ns/ft [4.1 ns/m]
Skew (within pair):	50 ps/meter maximum
Skew (channel skew per chipset):	50 ps/meter maximum

DataRate	Skew	Cable Length
40Mhz	390ps	7,8m
66MHz	290ps	5,8m
70MHz	270ps	5,4m
80MHz	218ps	4,36m
85MHz	190ps	3,8m

Starting with the firmware version 2.0.4B, the camera has been improved in term data output quality in order to push back the limitation and allow about 5m of length on some good cables (even specified at 50ps/m).

10 APPENDIX D: Lenses Compatibility

QIOPTICS (LINOS)

	Nominal Magnification	Magnification Range	M95 Focus tube Reference	Lens Reference Part number
Inspec.x. L 5.6/105	**0,33 X**	0,25 – 0,45 X	2408-012-000-41	0703-085-000-20
Inspec.x. L 5.6/105	**0,5 X**	0,4 – 0,65 X	2408-012-000-41	0703-084-000-20
Inspec.x. L 5.6/105	**0,87 X**	0,6 – 0,9 X	2408-012-000-43	0703-083-000-20
Inspec.x. L 5.6/105	**1 X**	0,85 – 1,2 X	2408-012-000-43	0703-082-000-20
Inspec.x. L 4/105	**3 X**	2,8 – 3,3 X	2408-012-000-46	0703-104-000-20
Inspec.x. L 4/105	**3,5 X**	3,3 – 3,7 X	2408-012-000-44	0703-095-000-21
Inspec.x. L 3.5/105	**5 X**	4,8 – 5,2 X	2408-012-000-45	0703-102-000-20

SCHNEIDER KREUZNACH

	Nominal Magnification	Magnification Range	Working Distance (at nom. Mag.)	Reference Part number
SR 5.6/120-0058	**1 X**	0,88 – 1,13 X	212 mm	1002647
SR 5.6/120-0059	**0,75 X**	0,63 – 0,88 X	252 mm	1002648
SR 5.6/120-0060	**0,5 X**	0,38 – 0,63 X	333 mm	1002650
SR 5.6/120-0061	**0,33 X**	0,26 – 0,38 X	453 mm	1004611
Accessories	V mount 25mm macro-extension tube		Necessary to combine the whole lens system	20179
	V mount to Leica adapter			20054
	Unifoc 76			13048
	Adapter M58x0.75 – M95x1			1062891
	Extension tube M95x1, 25mm		To be combined to reach the appropriate magnification	1062892
	Extension tube M95x1, 50mm			1062893
	Extension tube M95x1, 100mm			1062894

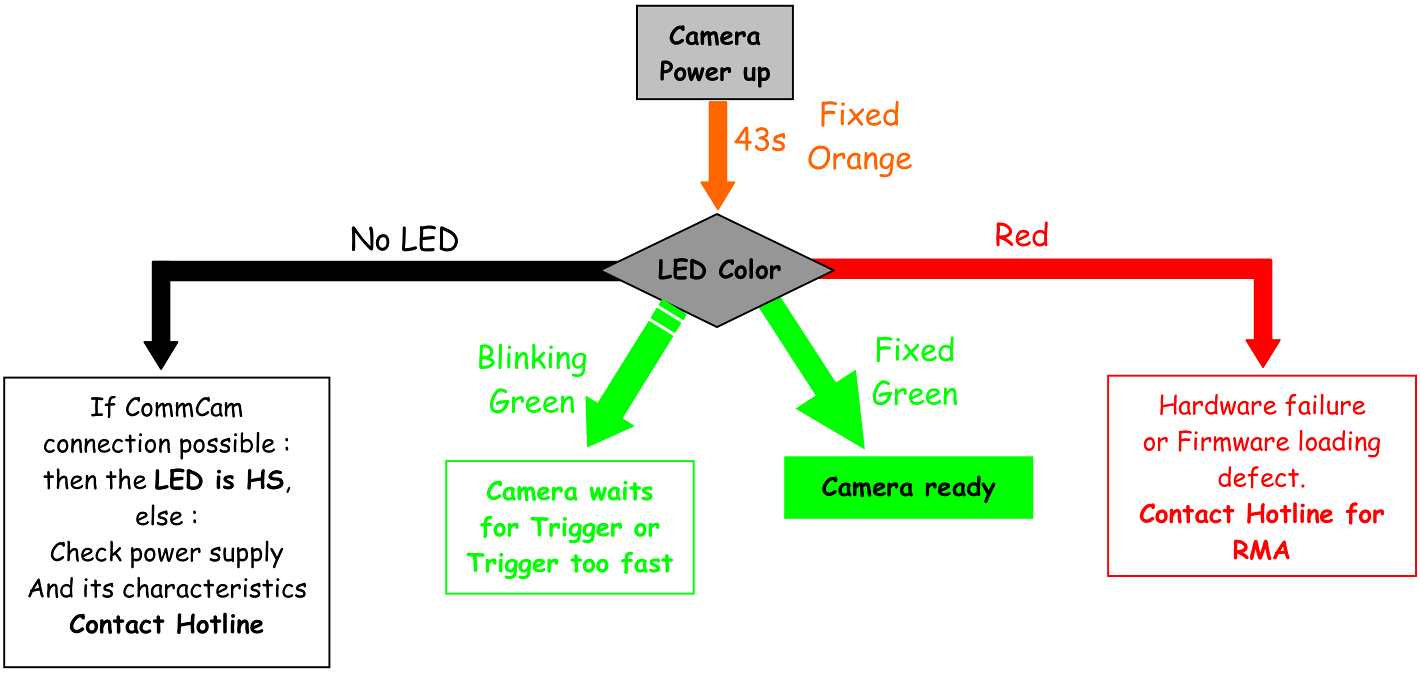
MYUTRON

	Nominal Magnification	Working Distance	
XLS03-E	**x0,3**	**477mm**	**M95 Custom Mount available** **Aperture (∞) : 4.7**
XLS53-E	**x0,5**	**324mm**	
XLS75-E	**x0,75**	**246mm**	
XLS010-E	**x1**	**197mm**	
XLS014-E	**x1,4**	**170mm**	
XLS203-E	**x2**	**146mm**	

EDMUND OPTICS			
	Nominal Magnification	**Working Distance**	**Reference**
TechSpec F4	**1 X**	151 mm	NT68-222
TechSpec F4	**1,33 X**	158,5 mm	NT68-223
TechSpec F4	**2,0 X**	129 mm	NT68-224
TechSpec F4	**3,0 X**	110 mm	NT68-225
Accessories	Large Format Tip/Tilt Bolt Pattern Adapter, 2X		NT69-235
	Large Format Focusing Module		NT69-240
	Large Format Adapter Set		NT69-241
NAVITAR			
Raptar Pro 4/86	**1 X**	Extension Tubes on request	1 - 17494
NIKON			
Rayfact F4	**0,05 X – 0,5 X**	1820,4mm – 230,3mm	Rayfact ML90mm F4

11 APPENDIX E: TROUBLESHOOTING

11.1 Camera



11.2 CommCam Connection

Refer to CommCam software Help for the connection issues.

12 APPENDIX F: Command Summary Tables

12.1 Device Control

Feature	CL Command	Description
DeviceVendorName	r vdnm	Get camera vendor name as a string (32 bytes long including '\0')
DeviceModelName	r mdnm	Get camera model name as a string (32 bytes long including '\0')
DeviceFirmwareVersion	r dfwv	Get camera synthetic firmware version (PKG version) as a string (32 bytes long including '\0')
DeviceVersion	r dhwv	Get camera version as a string (hardware version) (32 bytes long including '\0')
DeviceManufacturerInfo	r idnb	Get camera ID as a string (48 bytes long including '\0')
DeviceUserID	r cust	Get device user identifier as a string (16 bytes long including '\0')
	w cust <idstr>	Set camera identifier to <idstr>
DeviceID	r deid	Read Serial Nb
ElectronicBoardID	r boid	Read Electronic Board ID
DeviceSFNCVersionMajor	Xml Virtual	
DeviceSFNCVersionMinor	Xml Virtual	
DeviceSFNCVersionSubMinor	Xml Virtual	

12.2 Image Format

Feature	Command	Description
SensorWidth	r snsw	Get sensor physical width.
SensorHeight	Xml virtual	
WidthMax	Map on SensorWidth	
HeightMax	Xml virtual	
Height	Xml virtual	
Width	Xml virtual	Depends on (OuputRegion, OuputRegionWidth) and SensorWidth
SensorMode	r smod	Get sensor mode
	w smod 0	Set sensor mode to DualLine "1S
	w smod 1	Set sensor mode to MultiLine "2S
	w smod 2	Set sensor mode to QuadriLine "4S"
MultiLineGain	r mlig	Get MultiLine gain
	w mlig 0	Set MultiLine gain to "x1"
	w mlig 1	Set MultiLine gain to "x1/2" (not available if SensorMode = 0 or = 5 ("1S"/"4S" modes)
ScanDirection	r scdi	Get scan direction
	w scdi 0	Set scan direction to "forward"
	w scdi 1	Set scan direction to "reverse"
	w scdi 2	Set scan direction to "Externally controlled direction via CC3 Camera Link (CC3=0 forward, CC3=1 reverse)"
ReverseReading	r revr	Get reverse reading value

Feature	Command	Description
	w revr 0	Set reverse reading to "disable"
	w revr 1	Set reverse reading to "enable"
OutputMode	r mode	Get output mode (CameraLink configuration and CMOS sensor resolution)
	w mode 0	Set output mode to "Medium4Outputs8bits"
	w mode 1	Set output mode to "Medium4Outputs12bits"
	w mode 2	Set output mode to "Full8Outputs8bits"
	w mode 3	Set output mode to "FullPlus10Outputs8bits"
OutputFrequency	r clfq	Get Camera Link frequency
TestImageSelector	r srce	Get test (output FPGA) image pattern
	w srce 0	Set test (output FPGA) image pattern to "Off", processing chaine activated
	w srce 1	Set test (output FPGA) image pattern to "GreyHorizontalRamp", processing chaine desactivated
	w srce 2	Set test (output FPGA) image pattern to "White pattern", processing chaine desactivated
	w srce 3	Set test (output FPGA) image pattern to "gray pattern", processing chaine desactivated
	w srce 4	Set test (output FPGA) image pattern to "Black pattern", processing chaine desactivated
	w srce 5	Set test (output FPGA) image pattern to "GreyVerticalRampMoving", processing chaine desactivated

12.3 Synchro and Acquisition

Feature	Commands	Description
LinePeriod	r tper	Get current line period
	w tper <val>	Set line period, from from 1 (0,1µs) to 65535 (6553,5µs), step 1 (0,1µs)
LinePeriodMin	r tpmi	Get current line period min (0..65535 step 0,1µs)
AcquisitionLineRate	Xml Virtual	= 1 / LinePeriod en Hertz
ExposureTime~~Abs~~	r tint	Get exposure time
	w tint <val>	Set exposure time, from 1 (0,1µs) to 65535 (6553,5µs), step 1 (0,1µs)
TriggerPreset	r sync	Get trigger preset mode
	w sync 0	Set trigger preset mode to Freerun timed mode, with exposure time and line period programmable
	w sync 1	Set trigger preset mode to Triggered mode with exposure time settings
	w sync 2	Set trigger preset mode to Triggered mode with maximum exposure time
	w sync 3	Set trigger preset mode to Triggered mode with exposure time controlled by one signal
	w sync 4	Set trigger preset mode to Triggered mode with exposure time controlled by two signals
	w sync 5	Set trigger preset mode to Freerun mode, with max exposure time and programmable line period

12.4 Gain & Offset

Feature	Commands	Description
GainAbs **GainSelector= AnalogAll**	r pamp	Get the current pre-amp gain
	w pamp <val>	Set pre amplifier gain to: 0 (-12dB), 1 (-6dB), 2 (0dB) (analog gain) Change balances and compensation
GainAbs **GainSelector= gainAll**	r gain	Get current digital gain
	w gain <val>	Set gain from 0dB(0) to +8 dB (6193)
Gain Abs **GainSelector=DigitalAll**	r gdig	Get contrast expansion digital gain
	w gdig <val>	Set contrast expansion digital gain from 0 (0 dB) to 255 (+14 dB), step 1 (TBD dB)
BlackLevelRaw **BlackLevelSelector=All**	r offs	Get common black level.
	w offs <val>	Set common black from -4096 to 4095, step 1
GainAbs **GainSelector=DigitalTap<j>**	r fga<j> <val>	Get tap<j> digital gain. Dynamically updated on AnalogAll gain changes
	w fga<j> <val>	Set tap<j> digital gain from -128 to 127 by step 1 (0.0021dB). Dynamically updated on AnalogAll gain changes
ROI Gain Set	r roig	Read the last ROI gain set
	w roig <val>	Set the Value for the ROI Gain : <val> from 0 to 2047 : U1.11 (1+coeff/1024) => x1..x1.999877 step 1/1024
ROI Set	r rois	Read the last ROI set
	w rois <val>	Set the ROI and applies the ROI Gain on it. <val> is a combination of Start and Stop Addresses for ROI. - Start Address : From 0 to 16382 (0x0000 to 0x3FFE) - Stop Address : From 1 to 16383 (0x0001 to 0x3FFF)

12.5 Flat Field Correction

Feature	Commands	Description
FFCEnable	r ffcp	Get Flat Field Correction processing status
	w ffcp 0	Disable Flat Field Correction ("False")
	w ffcp 1	Enable Flat Field Correction ("True")
FPNReset	w rsto 0	Reset FPN coefficients
PRNUReset	w rstg 0	Reset PRNU coefficients
No direct feature	r ffco <addr>	Read 128 Fpn coefficients starting from address <addr>. Return value is in hexadecimal, without space between values (one unsigned short per coef). Format: S9.1 => -256..+255.5 step 1/2
	w ffco <addr> <val>	Write 128 Fpn coefficients (straight to FPGA) starting from address <addr>. <val> is the concatenation of individual Fpnvalue, without space between values.
No direct feature	r ffcg <addr>	Read 128 Prnu coefficients (straight from FPGA) starting from address <addr>. Return value is in hexadecimal, without space between values. Coeff from 0 to 4095 : U2.12 (1+coeff/1024) => x1..x4.999877 step 1/1024
	w ffcg <addr> <val>	Write 128 Prnu coefficients (straight to FPGA) starting from address <addr>. <val> is the concatenation of individual PRNUvalue, without space between values.
FFCCalibrationCtrl	r calg	Get the PRNU calibration status
	w calg 0	Abort PRNU calibration by setting it to "Off" (no effect if already stopped)
	w calg 1	Launch PRNU calibration by setting it to "Once" (no effect if already launched)

Feature	Commands	Description
PrnuCalibrationCtrl	r calo	Get the fpn calibration status
	w calo 0	Abort fpn calibration by setting it to "Off" (no effect if already stopped)
	w calo 1	Launch fpn calibration by setting it to "Once" (no effect if already launched)
FFCAdjust	r ffad	Get ffc adjust state
	w ffad 0	Disable ffc adjust
	w ffad 1	Enable ffc adjust
FFCAutoTargetLevel	r tfad	Get the FFC target adjust level
	W tfad <val>	Set FFC target adjust level, from 0 to 4095, step 1

12.6 LUT

Feature	Commands	Description
LUTEnable	r lute	Get LUT status
	w lute 0	Disable LUT ("False")
	w lute 1	Enable LUT ("True")
No direct feature	r lutc <addr>	Read 128 LUT coefficients starting from address <addr> from 0 to 4095-128. Return value is in hexadecimal, without space between values. (one unsigned char per coef)
	w lutc <addr> <val>	Write 128 LUT coefficients starting from address <addr> from 0 to 4095-128. <val> is the concatenation of individual LUTvalue, without space between values.

12.7 Save and Restore

Feature	Commands	Description
UserSetLoad	r rcfg	Get the current user configuration bank (saved or restored)
	w rcfg <val>	Restore current UserSet from UserSet bank number <val>, from 0 to 5; <val> comes from UserSetSelector.
UserSetSave	w scfg <val>	Save current UserSet to UserSet bank number <val>, from 1 to 5; <val> comes from UserSetSelector. 0 cannot be saved. 5 (Integrator) can't be saved in User mode
UserSetControl	Xml virtual	
RestoreLUTFromBank	r rlut	Get the current LUT bank (saved or restore)
	w rlut <val>	Restore current LUT from LUT bank number <val>, from 1 to 4; <val> comes from LUTSetSelector.
SaveLUTToBank	w slut <val>	Save current LUT to LUT FFC bank number <val>, from 1 to 4; <val> comes from LUTSetSelector.
RestoreFFCFromBank	r rffc	Get the current FFC bank (save or restore)
	w rffc <val>	Restore current FFC (including FPN and FFCGain) from FFC bank number <val>, from 1 to 4; <val> comes from UserFFCSelector (XML feature).
SaveFFCToBank	w sffc <val>	Save current FFC (including FPN and FFCGain) to FFC bank number <val>, from 1 to 4; <val> comes from FFCSelector (XML feature).

12.8 CAMERA STATUS

Feature	Commands	Description
PrivilegeLevel	r lock	Get camera running privilege level 0 = Privilege Factory 1 = Privilege Advanced User 2 = Privilege User
ChangePrivilegeLevel	w lock 1	Lock camera privilege to "Advanced User"
	w lock 2	Lock camera privilege to "User"
	w lock <val>	Unlock camera privilege depending on <val> (min=256; max=2^{32}-1)
DeviceTemperature	r temp	Read Mainboard internal temperature (format signed Q10.2 = signed 8 bits, plus 2 bits below comma. Value from -512 to +511) in °C
DeviceTemperatureSelector	Xml Virtual	
Standby	r stby	Read Standby state (CMOS sensor)
	w stby 0	Disable standby mode ("False")
	w stby 1	Enable standby mode ("True"), no more video available but save power and temperature
	r stat	Get camera status (see below for details)
StatusWaitForTrigger		Bit 0: true if camera waits for a trigger during more than 1s
Satus trigger too fast		Bit 1: true if camera trigger is too fast
StatusWarningOverflow		Bit 8: true if a an overflow occurs during FFC calibration or Tap balance (available only for integrator/user mode)
StatusWarningUnderflow		Bit 9: true if a an underflow occurs during FFC calibration or Tap balance (available only for integrator/user mode)
Cc3 Scrolling direction		Bit 11: 0 : forward, 1: reverse
StatusErrorHardware		Bit 16 : true if hardware error detected

12.9 communication

Feature	Commands	Description
ComBaudRate	r baud	Get current baud rate (This feature is not saved in camera)
	w baud 1	Set baud rate to "9600Bds"
	w baud 2	Set baud rate to "19200Bds"
	w baud 6	Set baud rate to "57600Bds"
	w baud 12	Set baud rate to "115200Bds"

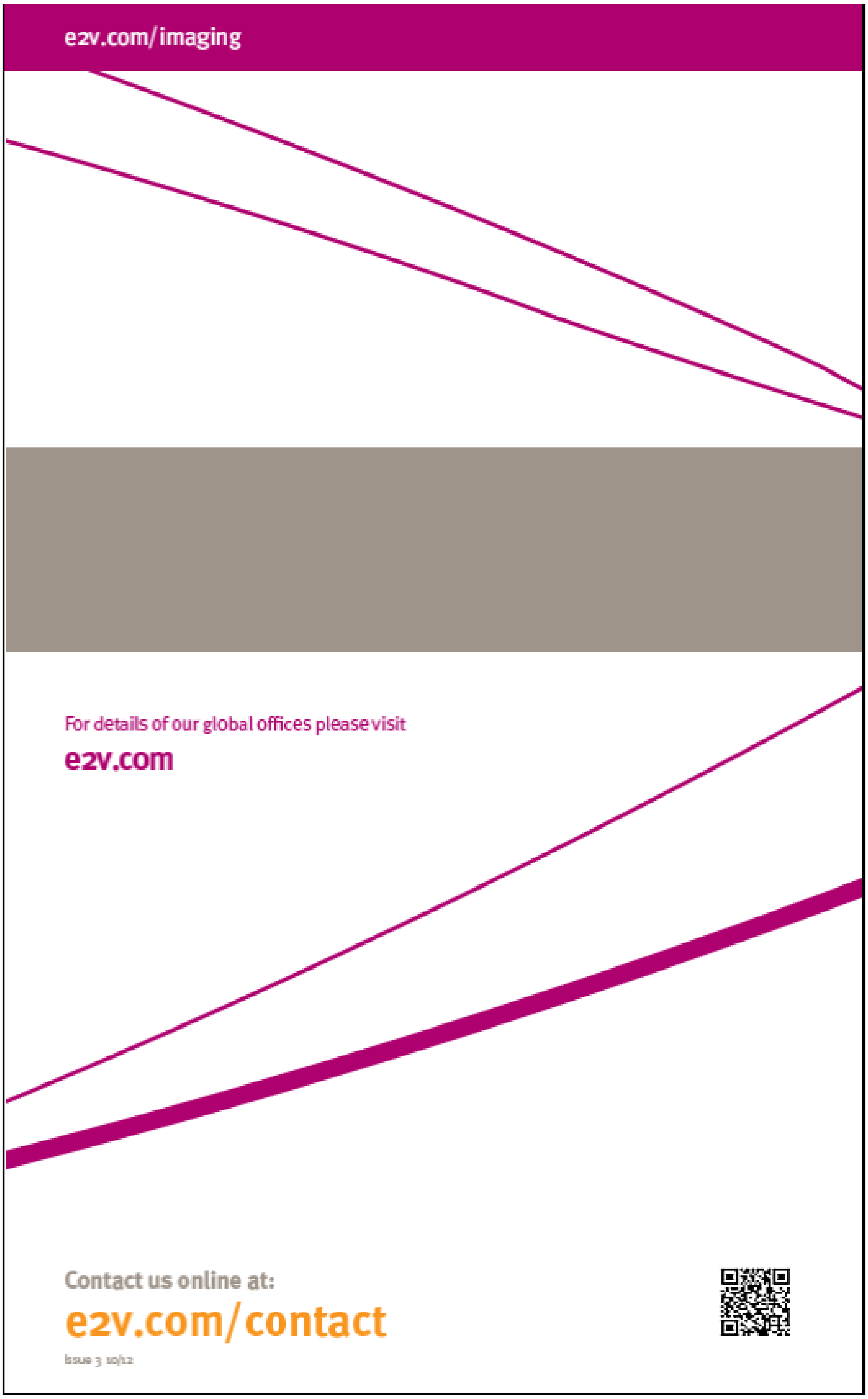
12.10 Line Profile Average

Feature	Commands	Description
LineAverageProfile	r pixs	Get the line Line Average Profile status - 1 : running - 0 : finished
	w pixs 0	Abort the Line Average Profile
	w pixs 1	Run the Line Average Profile
PixelAccessLineNumer	r pixl	Get the number of line for average
	w pixl <val>	Set the number of line to accumulate - <val> : 1,256,512,1024
No direct feature	r pixv <addr>	Read 128 pixel values starting from address <addr>, from SensorWidth-128-1. Return value is in hexadecimal, without space between values. (one unsigned short per coef)

Feature	Commands	Description
PixelRoiStart	r prod	Get Roi start
	w prod <val>	Set Roi start for pixel statistic computing (0 to SensorWidth -1-1)
PixelRoiWidth	r prow	Get Roi width
	W prow <val>	Set Roi width for pixel statistic computing (1 to SensorWidth)
PixelROIMean	r pavr	Get ROI Mean (format U12.4)
PixelROIStandardDeviation	r pstd	Get ROI Stand deviation (format U12.4)
PixelROIMin	r pmin	Get ROI Min (format U12.4)
PixelROIMax	r pmax	Get ROI Max (format U12.4)

13 APPENDIX G: Revision History

Manual Revision	Comments / Details	Firmware version	1st CommCam compliant Version
Rev A	First release	1.0.19	**2.1.0**
Rev B	Timing values and Electro-Optic performances	1.0.19	**2.1.0**
Rev C	EV71YC4MCL1605-BA1 : New sensor version - Full+ 10 Taps output Mode - 2x Binning modes - New Timing values for the synchro modes EV71YC4MCL1605-BA0 : no more updated.	2.x.x 1.0.19	**2.2.0** **2.1.0**
Rev D	Full Exposure Control in 4S mode (BA1 only)	2.0.4A	**2.1.0**
Rev E	Improved Data Output for 85MHz cables	2.0.4B	**2.1.3**
Rev F	Quarter Balance Gains	2.0.5A	**2.1.5**
Rev G	FFC Gain Range and ROI Gain	2.0.6A	**2.1.7**
Rev H	Characterization of Forward / Reverse	2.0.11	**2.1.7**


e2v.com/imaging
For details of our global offices please visit
e2v.com
Contact us online at:
e2v.com/contact
Issue 3 10/12